MicroStrategy | 10

Supplemental Reference for System Administration

Version: 10.8

10.8, June 2017

CONTENTS

Overview and Additional Resources **6**

What's new in this guide 7

Resources 8

Feedback 16

1. SQL Generation and Data Processing: VLDB Properties ***17***

Supporting your system configuration 18

Accessing and working with VLDB properties 19

Details for all VLDB properties 27

Default VLDB settings for specific data sources 225

2. Creating a Multilingual Environment: Internationalization ***228***

About internationalization 229

Best practices for implementing internationalization 231

Preparing a project to support internationalization 232

Providing metadata internationalization 235

Providing data internationalization 244

Making translated data available to users 251

Achieving the correct language display 265

Maintaining your internationalized environment 269

3. Multi-Tenant Environments: Object Name Personalization ***280***

How a tenant language differs from a standard language 280

Granting user access to rename objects and view tenant languages 281

Renaming metadata objects 282

Making tenant-specific data available to users 289

Achieving the correct language display 303

Maintaining your multi-tenant environment 305

4. List of Privileges 307

Privileges for predefined security roles 308
Privileges for out-of-the-box user groups 309
List of privileges required for specific Flash Mode interactions 314
List of all privileges 315

5. Command Manager Runtime 334

Statement Reference Guide 334
Executing a script with Command Manager Runtime 334
Syntax reference guide 336

6. Intelligence Server Statistics Data Dictionary 337

STG_CT_DEVICE_STATS 337
STG_CT_EXEC_STATS 339
STG_CT_MANIP_STATS 345
IS_ALERT_STATS 349
STG_IS_CACHE_HIT_STATS 352
STG_IS_CUBE_REP_STATS 354
STG_IS_DOC_STEP_STATS 357
STG_IS_DOCUMENT_STATS 362
STG_IS_INBOX_ACT_STATS 367
STG_IS_MESSAGE_STATS 371
STG_IS_PERF_MON_STATS 376
STG_IS_PR_ANS_STATS 378
STG_IS_PROJ_SESS_STATS 381
STG_IS_REP_COL_STATS 385
STG_IS_REP_SEC_STATS 387
STG_IS_REP_SQL_STATS 389
STG_IS_REP_STEP_STATS 395
STG_IS_REPORT_STATS 401
STG_IS_SCHEDULE_STATS 408
STG_IS_SESSION_STATS 410
STG_MSI_STATS_PROP 414

7. Enterprise Manager Data Model and Object Definitions 415

Enterprise Manager data warehouse tables 415
Relationship tables 455
Enterprise Manager metadata tables 456

Enterprise Manager attributes and metrics 457

8. MicroStrategy Web Cookies 867

Session information 867
Default user name 870
Project information 870
Current language 871
GUI settings 871
Personal autostyle information 871
System autostyle information 872
Connection information 872
Available projects information 872
Global user preferences 873
Cached preferences 873
Preferences 874

Index 879

OVERVIEW AND ADDITIONAL RESOURCES

This guide is a supplemental resource for system administrators when implementing, deploying, maintaining, tuning, and troubleshooting a MicroStrategy business intelligence system.

For detailed information about common system administration tasks, see the System Administration Guide.

The chapters provide the following information:

- *SQL Generation and Data Processing: VLDB Properties*

  This chapter shows you how to set the VLDB properties for your system, and provides a complete listing of all MicroStrategy VLDB properties.

- *Chapter 2, Creating a Multilingual Environment: Internationalization*

  This chapter shows you how to use MicroStrategy to internationalize a project in your MicroStrategy environment, to make it available to a multi-lingual audience. This includes internationalizing data in your data warehouse and metadata objects in the MicroStrategy metadata repository.

- *Chapter 3, Multi-Tenant Environments: Object Name Personalization*

  This chapter shows you how to use MicroStrategy to personalize object names in a project in your MicroStrategy environment, to support a multi-tenant setup.

- *Chapter 4, List of Privileges*

  This chapter provides a complete list of the privileges assigned to each predefined user group and security role. It also provides a complete list of all privileges available in MicroStrategy.

- *Chapter 5, Command Manager Runtime*

  This chapter provides a complete list of the commands available in Command Manager Runtime, a lightweight version of Command Manager designed to be bundled with OEM applications.

- *Chapter 6, Intelligence Server Statistics Data Dictionary*

This chapter provides a complete list of all the database tables used in the Intelligence Server statistics.

- *Chapter 7, Enterprise Manager Data Model and Object Definitions*

  This chapter provides a complete list of all the database tables and object definitions used in the Enterprise Manager project.

- *Chapter 8, MicroStrategy Web Cookies*

  This chapter provides a complete list of all the cookies used by MicroStrategy Web.

The sample documents and images in this guide, as well as some example steps, were created with dates that may no longer be available in the MicroStrategy Tutorial project. If you are re-creating an example, replace the year(s) shown in this guide with the most recent year(s) available in the software.

What's new in this guide

MicroStrategy 10.8

- To enable persistent quick search functionality in projects, the Change Journal functionality is now enabled by default.
- For MDX metric values, you can now select whether to format per column or cell. For more informatoin, see *MDX Cell Formatting*.

MicroStrategy 10.7

- When performing In-Memory calculations - such as derived metrics, subtotals, and dynamic aggregation - on simple and aggregation functions, we would ignore nulls during the calculation of aggregation functions while still treating them as 0 for scalar operations. The new calculation behavior is applied by default but can be reverted by through the Advanced VLDB Properties Editor. For more information, see *NULL checking for Analytical Engine*
- When running a multi-source report in 10.7, you can execute multiple report pre- and post-SQL statements in an expected order for both primary and all secondary databases. This can be helpful to database administrators who want to collect database statistics. The SQL statements are also executed when the report is used in a document or dashboard. To apply pre- and post-SQL statements, use the Pre/Post Statements VLDB setting. For more information on these SQL statements, see *Multi-source Report Pre and Post Statements*.

MicroStrategy 10.6

- Pre/Post Statements now support using **!f** to insert the full path to the report. For more information, see *Customizing SQL statements: Pre/Post Statements*.

MicroStrategy 10.0

- Documentation on new or modified VLDB properties:
  - *Modifying third-party cube sources in MicroStrategy: MDX, page 85*
  - *Limiting report rows, SQL size, and SQL time-out: Governing, page 49*
  - *Modifying third-party cube sources in MicroStrategy: MDX, page 85*
  - *Modifying third-party cube sources in MicroStrategy: MDX, page 85*
  - *Calculating data: Metrics, page 97*
  - *Calculating data: Metrics, page 97*
  - *Selecting and inserting data with SQL: Select/Insert, page 185*
  - *Selecting and inserting data with SQL: Select/Insert, page 185*
  - *Parallel SQL Execution Intermediate Table Type, page 224*

Resources

This section provides details on how to access books, online help, MicroStrategy Education and Consulting resources, and how to contact MicroStrategy Technical Support.

Documentation

MicroStrategy provides both manuals and online help; these two information sources provide different types of information, as described below:

- **Manuals**: MicroStrategy manuals provide:
  - Introductory information and concepts
  - Examples and images
  - Checklists and high-level procedures to get started

  The steps to access the manuals are described in *Accessing manuals and other documentation sources, page 13*.

  Most of these manuals are also available printed in a bound, soft cover format. To purchase printed manuals, contact your MicroStrategy Account Executive with a purchase order number.
- **Help**: MicroStrategy online help provides:
  - Detailed steps to perform procedures
  - Descriptions of each option on every software screen

Additional formats

MicroStrategy manuals are available as electronic publications, downloadable on the Apple iBooks Store or Google Play, and can be read on your iOS or Android device respectively. To download a book, search for the book's title in the iBookstore or Google Play. To view a list of manuals that are currently available, scan the following QR codes using your device's camera:

- For iOS devices, scan the following QR code:

- For Android devices, scan the following QR code:

For new MicroStrategy releases, it may take several days for the latest manuals to be available on the iBookstore or Google Play.

Translations

For the most up-to-date translations of MicroStrategy documentation, refer to the MicroStrategy Knowledge Base. Due to translation time, manuals in languages other than English may contain information that is one or more releases behind. You can see the version number on the title page of each manual.

Finding information

You can search all MicroStrategy books and Help for a word or phrase, with a simple Google™ search at *http://www.google.com*. For example, type "MicroStrategy derived metric" or "MicroStrategy logical table" into a Google search. As described above, books typically describe general concepts and examples; Help typically provides detailed steps and screen options. To limit your search to MicroStrategy books, on Google's main page you can click **More**, then select **Books**.

Manuals for MicroStrategy overview and evaluation

- *Introduction to MicroStrategy: Evaluation Guide*

  Instructions for installing, configuring, and using the MicroStrategy Evaluation Edition of the software. This guide includes a walkthrough of MicroStrategy features so you can perform reporting with the MicroStrategy Tutorial project and its sample business data.

- *MicroStrategy Evaluation Edition Quick Start Guide*

  Overview of the installation and evaluation process, and additional resources.

Resources for security

- *Usher Help*

  Steps to perform mobile identity validation using the Usher mobile security network to issue electronic badges for identifying users.

Manuals for query, reporting, and analysis

- *MicroStrategy Installation and Configuration Guide*

  Information to install and configure MicroStrategy products on Windows, UNIX, Linux, and HP platforms, and basic maintenance guidelines.

- *MicroStrategy Upgrade Guide*

  Steps to upgrade existing MicroStrategy products.

- *MicroStrategy Project Design Guide*

  Information to create and modify MicroStrategy projects, and create the objects that present your organization's data, such as facts, attributes, hierarchies, transformations, advanced schemas, and project optimization.

- *MicroStrategy Basic Reporting Guide*

  Steps to get started with MicroStrategy Web, and how to analyze and format data in a report. Includes the basics for creating reports, metrics, filters, and prompts.

- *MicroStrategy Advanced Reporting Guide: Enhancing Your Business Intelligence Application*

  Steps to create Freeform SQL reports, Query Builder reports, complex filters and metrics, use Data Mining Services, and create custom groups, consolidations, and complex prompts.

- *MicroStrategy Report Services Document Creation Guide: Creating Boardroom Quality Documents*

Steps to create Report Services documents, add objects, and format the document and its objects.

- *MicroStrategy Dashboards and Widgets Creation Guide: Creating Interactive Dashboards for Your Data*

  Steps to create MicroStrategy Report Services dashboards and add interactive visualizations.

- *MicroStrategy In-memory Analytics Guide*

  Information to use MicroStrategy OLAP Services features, including Intelligent Cubes, derived metrics, derived elements, dynamic aggregation, view filters, and dynamic sourcing.

- *MicroStrategy Office User Guide*

  Instructions to use MicroStrategy Office to work with MicroStrategy reports and documents in Microsoft® Excel, PowerPoint, and Word, to analyze, format, and distribute business data.

- *MicroStrategy Mobile Analysis Guide: Analyzing Data with MicroStrategy Mobile*

  Steps to use MicroStrategy Mobile to view and analyze data, and perform other business tasks with MicroStrategy reports and documents on a mobile device.

- *MicroStrategy Mobile Design and Administration Guide: A Platform for Mobile Intelligence*

  Information and instructions to install and configure MicroStrategy Mobile, as well as steps for a designer working in MicroStrategy Developer or MicroStrategy Web to create effective reports and documents for use with MicroStrategy Mobile.

- *MicroStrategy System Administration Guide: Tuning, Monitoring, and Troubleshooting Your MicroStrategy Business Intelligence System*

  Steps to implement, deploy, maintain, tune, and troubleshoot a MicroStrategy business intelligence system.

- *MicroStrategy Supplemental Reference for System Administration: VLDB Properties, Internationalization, User Privileges, and other Supplemental Information for Administrators*

  Steps for administrative tasks such as configuring VLDB properties and defining data and metadata internationalization, and reference material for other administrative tasks.

- *MicroStrategy Functions Reference*

  Function syntax and formula components; instructions to use functions in metrics, filters, attribute forms; examples of functions in business scenarios.

- *MicroStrategy MDX Cube Reporting Guide*

Information to integrate MicroStrategy with MDX cube sources. You can integrate data from MDX cube sources into your MicroStrategy projects and applications.

- *MicroStrategy Operations Manager Guide*

  Instructions for managing, monitoring, and setting alerts for all of your MicroStrategy systems from one console. This guide also includes instructions for setting up and using Enterprise Manager to analyze your MicroStrategy system usage.

Manual for the Human Resources Analytics Module

- *Human Resources Analytics Module Reference*

Software Development Kits

- *MicroStrategy Developer Library (MSDL)*

  Information to understand the MicroStrategy SDK, including details about architecture, object models, customization scenarios, code samples, and so on.

- *MicroStrategy Web SDK*

  The Web SDK is available in the MicroStrategy Developer Library, which is part of the MicroStrategy SDK.

Documentation for MicroStrategy Portlets

- *Enterprise Portal Integration Help*

  Information to help you implement and deploy MicroStrategy BI within your enterprise portal, including instructions for installing and configuring out-of-the-box MicroStrategy Portlets for several major enterprise portal servers.

  This resource is available from *http://www.microstrategy.com/producthelp.*

Documentation for MicroStrategy GIS Connectors

- *GIS Integration Help*

  Information to help you integrate MicroStrategy with Geospatial Information Systems (GIS), including specific examples for integrating with various third-party mapping services.

  This resource is available from *http://www.microstrategy.com/producthelp.*

Help

Each MicroStrategy product includes an integrated help system to complement the various interfaces of the product as well as the tasks that can be accomplished using the product.

Some of the MicroStrategy help systems require a web browser to be viewed. For supported web browsers, see the MicroStrategy Readme.

MicroStrategy provides several ways to access help:

- Help button: Use the Help button or ? (question mark) icon on most software windows to see help for that window.
- Help menu: From the Help menu or link at the top of any screen, select MicroStrategy Help to see the table of contents, the Search field, and the index for the help system.
- F1 key: Press F1 to see context-sensitive help that describes each option in the software window you are currently viewing.

For MicroStrategy Web, MicroStrategy Web Administrator, and MicroStrategy Mobile Server, pressing the F1 key opens the context-sensitive help for the web browser you are using to access these MicroStrategy interfaces. Use the Help menu or ? (question mark) icon to access help for these MicroStrategy interfaces.

Accessing manuals and other documentation sources

The manuals are available from http://www.microstrategy.com/producthelp, as well as from your MicroStrategy disk or the machine where MicroStrategy was installed.

Adobe Reader is required to view these manuals. If you do not have Adobe Reader installed on your computer, you can download it from http://get.adobe.com/reader/.

The best place for all users to begin is with the *MicroStrategy Basic Reporting Guide*.

To access the installed manuals and other documentation sources, see the following procedures:

- *To access documentation resources from any location, page 13*
- *To access documentation resources on Windows, page 14*
- *To access documentation resources on UNIX and Linux , page 14*

To access documentation resources from any location

1 Visit *http://www.microstrategy.com/producthelp*.

To access documentation resources on Windows

1. From the Windows **Start** menu, choose **Programs** (or **All Programs**), **MicroStrategy Documentation**, then **Product Manuals**. A page opens in your browser showing a list of available manuals in PDF format and other documentation sources.
2. Click the link for the desired manual or other documentation source.

If bookmarks are not visible on the left side of a product manual, from the **View** menu click **Bookmarks and Page**. This step varies slightly depending on your version of Adobe Reader.

To access documentation resources on UNIX and Linux

1. Within your UNIX or Linux machine, navigate to the directory where you installed MicroStrategy. The default location is `/opt/MicroStrategy`, or `$HOME/MicroStrategy/install` if you do not have write access to `/opt/MicroStrategy`.
2. From the MicroStrategy installation directory, open the `Help` folder.
3. Open the `Product_Manuals.htm` file in a web browser. A page opens in your browser showing a list of available manuals in PDF format and other documentation sources.
4. Click the link for the desired manual or other documentation source.

If bookmarks are not visible on the left side of a product manual, from the **View** menu click **Bookmarks and Page**. This step varies slightly depending on your version of Adobe Reader.

Documentation standards

MicroStrategy online help and PDF manuals (available both online and in printed format) use standards to help you identify certain types of content. The following table lists these standards.

These standards may differ depending on the language of this manual; some languages have rules that supersede the table below.

Type	Indicates
bold	• Button names, check boxes, options, lists, and menus that are the focus of actions or part of a list of such GUI elements and their definitions Example: Click **Select Warehouse**.

Type	Indicates
italic	• Names of other product manuals and documentation resources • When part of a command syntax, indicates variable information to be replaced by the user Example: Type `copy c:\`*filename* `d:\`*foldername*`\`*filename*
`Courier font`	• Calculations • Code samples • Registry keys • Path and file names • URLs • Messages displayed in the screen • Text to be entered by the user Example: `Sum(revenue)/number of months.` Example: Type `cmdmgr -f scriptfile.scp` and press **Enter**.
+	A keyboard command that calls for the use of more than one key (for example, SHIFT+F1).
	A note icon indicates helpful information for specific situations.
	A warning icon alerts you to important information such as potential security risks; these should be read before continuing.

Education

MicroStrategy Education Services provides a comprehensive curriculum and highly skilled education consultants. Many customers and partners from over 800 different organizations have benefited from MicroStrategy instruction.

Courses that can help you prepare for using this manual or that address some of the information in this manual include:

- Implementing MicroStrategy: Development and Deployment
- MicroStrategy Administration

For the most up-to-date and detailed description of education offerings and course curricula, visit *http://www.microstrategy.com/Education*.

Consulting

MicroStrategy Consulting Services provides proven methods for delivering leading-edge technology solutions. Offerings include complex security architecture designs, performance and tuning, project and testing strategies and recommendations, strategic planning, and more. Click here for a detailed description of consulting offerings.

Technical Support

If you have questions about a specific MicroStrategy product, you should:

1. Consult the product guides, Help, and readme files. Locations to access each are described above.
2. Consult the MicroStrategy Knowledge Base online at *https://resource.microstrategy.com/support*.

   A technical administrator in your organization may be able to help you resolve your issues immediately.

3. MicroStrategy Technical Support can be contacted by your company's Support Liaison. Contact information and the Technical Support policy information is available at *http://www.microstrategy.com/services-support/support/contact*.

Feedback

(missing or bad snippet)

1

SQL Generation and Data Processing: VLDB Properties

VLDB properties allow you to customize the SQL that MicroStrategy generates, and determine how data is processed by the Analytical Engine. You can configure properties such as SQL join types, SQL inserts, table creation, Cartesian join evaluation, check for null values, and so on.

VLDB properties can provide support for unique configurations and optimize performance in special reporting and analysis scenarios. You can use the VLDB Properties Editor to alter the syntax or behavior of a SQL statement and take advantage of unique, database-specific optimizations. You can also alter how the Analytical Engine processes data in certain situations, such as subtotals with consolidations and sorting null values.

Each VLDB property has two or more VLDB settings which are the different options available for a VLDB property. For example, the Metric Join Type VLDB property has two VLDB settings, Inner Join and Outer Join.

Some of the qualities that make VLDB properties valuable are:

- Complete database support: VLDB properties allow you to easily incorporate and take advantage of new database platforms and versions.
- Optimization: You can take advantage of database-specific settings to further enhance the performance of queries.
- Flexibility: VLDB properties are available at multiple levels so that the SQL generated for one report, for example, can be manipulated separately from the SQL generated for another, similar report. For a diagram, see *Order of precedence, page 18*.

Modifying any VLDB property should be performed with caution only after understanding the effects of the VLDB settings you want to apply. A given VLDB setting can support or optimize one system setup, but the same setting can cause performance issues or errors for other systems. Use this manual to learn about the VLDB properties before modifying any default settings.

Supporting your system configuration

Different SQL standards among various database platform (DBMS) types require that some VLDB properties are initialized to different default settings depending on the DBMS used. For example, when using a Microsoft Access 2000 database, the Join Type VLDB property is set to Join 89. This type of initialization ensures that different DBMS types can be supported. These initializations are also used as the default VLDB settings for the respective DBMS type. To create and review a detailed list of all the default VLDB settings for different DBMS types, see *Default VLDB settings for specific data sources, page 225*.

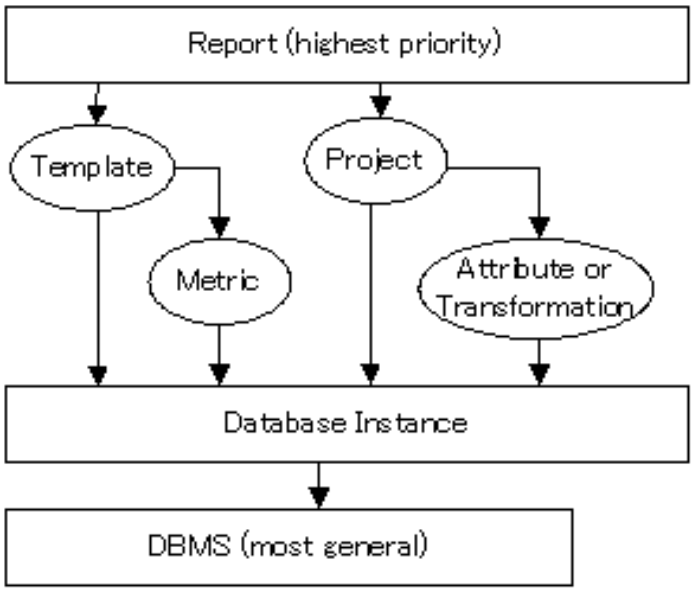
VLDB properties also help you configure and optimize your system. You can use MicroStrategy for different types of data analysis on a variety of data warehouse implementations. VLDB properties offer different configurations to support or optimize your reporting and analysis requirements in the best way.

For example, you may find that enabling the Set Operator Optimization VLDB property provides a significant performance gain by utilizing set operators such as `EXCEPT` and `INTERSECT` in your SQL queries. On the other hand, this property must offer the option to be disabled, since not all DBMS types support these types of operators. VLDB properties offer you a choice in configuring your system.

Order of precedence

VLDB properties can be set at multiple levels, providing flexibility in the way you can configure your reporting environment. For example, you can choose to apply a setting to an entire database instance or only to a single report associated with that database instance.

The following diagram shows how VLDB properties that are set for one level take precedence over those set for another.



The arrows depict the override authority of the levels, with the report level having the greatest authority. For example, if a VLDB property is set one way for a report and the same property is set differently for the database instance, the report setting takes precedence.

Properties set at the report level override properties at every other level. Properties set at the template level override those set at the metric level, the database instance level, and the DBMS level, and so on.

A limited number of properties can be applied at each level.

Accessing and working with VLDB properties

Opening the VLDB Properties Editor

You can change the VLDB settings for different levels using the VLDB Properties Editor. (Levels are described in *Order of precedence, page 18*.) You can access the VLDB Properties Editor in several ways, depending on what level of MicroStrategy objects you want to impact with your VLDB property changes. For example, you can apply a setting to an entire database instance, or only to a single report associated with that database instance.

When you access the VLDB Properties Editor for a database instance, you see the most complete set of the VLDB properties. However, not all properties are available at the database instance level. The rest of the access methods have a limited number of properties available depending on which properties are supported for the selected object/level.

The table below describes every way to access the VLDB Properties Editor:

To set VLDB properties at this level	Open the VLDB Properties Editor this way
Attribute	In the Attribute Editor, on the **Tools** menu, select **VLDB Properties**.
Database Instance	Choose one of the following: • In the Database Instance Manager, right-click the database instance you want to modify VLDB settings for, and choose **VLDB Properties**. • In the Project Configuration Editor, select the **Database Instances: SQL data warehouses** or the **Database Instances: MDX data warehouses** category, then click **VLDB Properties**.
Metric	In the Metric Editor, on the **Tools** menu, point to **Advanced Settings**, and then select **VLDB Properties**.
Project	In the Project Configuration Editor, expand **Project definition**, and select **Advanced**. In the Project-Level VLDB settings area, click **Configure**.
Report (or Intelligent Cube)	In the Report Editor or Report Viewer, on the **Data** menu, select **VLDB Properties**. This is also the location in which you can access the VLDB Properties Editor for

To set VLDB properties at this level	Open the VLDB Properties Editor this way
	Intelligent Cubes.
Template	In the Template Editor, on the **Data** menu, select **VLDB Properties**.
Transformation	In the Transformation Editor, on the **Tools** menu, select **VLDB Properties**. Only one property (Transformation Role Processing) is available at this level. All other VLDB properties must be accessed from one of the other levels listed in this table.

- Only a single property, called Unbalanced or Ragged Hierarchy, can be set at the hierarchy level. This property's purpose and instructions to set it are described in the *MicroStrategy MDX Cube Reporting Guide*.
- VLDB properties exist at the filter level and the function level, but they are not accessible through the VLDB Properties Editor.
- All VLDB properties at the DBMS level are used for initialization and debugging only. You cannot modify a VLDB property at the DBMS level.

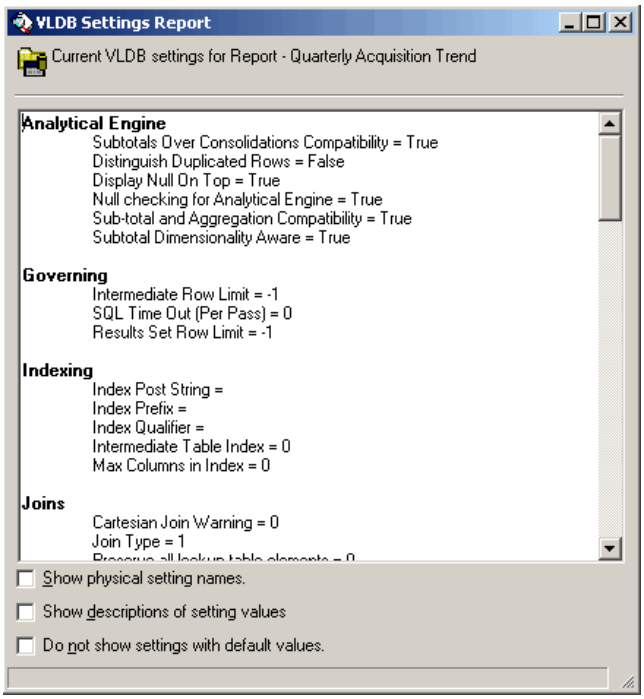
The VLDB Properties Editor has the following areas:

- **VLDB Settings list**: Shows the list of folders into which the VLDB properties are grouped. Expand a folder to see the individual properties. The settings listed depend on the level at which the VLDB Properties Editor was accessed (see the table above). For example, if you access the VLDB Properties Editor from the project level, you only see Analytical Engine properties.
- **Options and Parameters** box: Where you set or change the parameters that affect the SQL syntax.
- **SQL preview** box: (Only appears for VLDB properties that directly impact the SQL statement.) Shows a sample SQL statement and how it changes when you edit a property.

When you change a property from its default, a check mark appears on the folder in which the property is located and on the property itself.

Creating a VLDB settings report

A VLDB settings report displays all the current settings for each VLDB property that is available through a given instance of the VLDB Properties Editor. Part of a sample report of settings is shown below for VLDB properties available at the report level:



For each report, you can also decide whether to:

- Display the physical setting names alongside the names that appear in the interface. The physical setting names can be useful when you are working with MicroStrategy Technical Support to troubleshoot the effect of a VLDB property.
- Display descriptions of the values for each setting. This displays the full description of the option chosen for a VLDB property.
- Hide all settings that are currently set to default values. This can be useful if you want to see only those properties and their settings which have been changed from the default.

The steps below show you how to create a VLDB settings report. A common scenario for creating a VLDB settings report is to create a list of default VLDB settings for the database or other data source you are connecting to, which is described in *Default VLDB settings for specific data sources, page 225*.

To create a VLDB settings report

1. Open the VLDB Properties Editor to display the VLDB properties for the level at which you want to work. (For information on accessing the VLDB Properties Editor, see *Opening the VLDB Properties Editor, page 19*.)
2. From the **Tools** menu, select **Create VLDB Settings Report**.
3. A report is generated that displays all VLDB properties available at the level from which you accessed the VLDB Properties Editor. It also displays all current settings for each VLDB property.

4 You can choose to have the report display or hide the information described above, by selecting the appropriate check boxes.

5 You can copy the content in the report using the Ctrl+C keys on your keyboard. Then paste the information into a text editor or word processing program (such as Microsoft Word) using the Ctrl+V keys.

Viewing and changing VLDB properties

You can change VLDB properties to alter the syntax of a SQL statement and take advantage of database-specific optimizations.

Modifying any VLDB property should be performed with caution only after understanding the effects of the VLDB settings that you want to apply. A given VLDB setting can support or optimize one system setup, but the same setting can cause performance issues or errors for other systems. Use this manual to learn about the VLDB properties before modifying any default settings.

Some VLDB properties are characterized as "advanced properties": advanced properties are relevant only to certain projects and system configurations. To work with advanced VLDB properties, see *Viewing and changing advanced VLDB properties, page 22*.

To view and change VLDB properties

1 Open the VLDB Properties Editor to display the VLDB properties for the level at which you want to work. (For information on object levels, see *Order of precedence, page 18*.)

2 Modify the VLDB property you want to change. For use cases, examples, sample code, and other information on every VLDB property, see *Details for all VLDB properties, page 27*.

3 If necessary, you can ensure that a property is set to the default. At the bottom of the Options and Parameters area for that property (on the right), select the **Use default inherited value** check box. Next to this check box name, information appears about what level the setting is inheriting its default from.

4 Click **Save and Close** to save your changes and close the VLDB Properties Editor.

5 You must also save in the object or editor window through which you accessed the VLDB Properties Editor. For example, if you accessed the VLDB properties by opening the Metric Editor and then opening the VLDB Properties Editor, after you click **Save and Close** in the VLDB Properties Editor, you must also click **Save and Close** in the Metric Editor to save your changes to VLDB properties.

Viewing and changing advanced VLDB properties

By default, some VLDB properties are hidden when you open the VLDB Properties Editor. These properties are categorized as advanced VLDB properties because in general they are used infrequently and are relevant to only certain projects and system configurations. These settings are not dependent on any user privileges.

When modifying advanced VLDB properties, the same caution should be taken as when modifying any other VLDB property.

To display the advanced properties

1. Open the VLDB Properties Editor to display the VLDB properties for the level at which you want to work. (For information on object levels, see *Order of precedence, page 18*.)
2. From the **Tools** menu, select **Show Advanced Settings**. All advanced properties display with the other properties.
3. Modify the VLDB property you want to change. For use cases, examples, sample code, and other information on every VLDB property, see *Details for all VLDB properties, page 27*.
4. If necessary, you can ensure that a property is set to the default. At the bottom of the Options and Parameters area for that property (on the right), select the **Use default inherited value** check box. Next to this check box name, information appears about what level the setting is inheriting its default from.
5. Click **Save and Close** to save your changes and close the VLDB Properties Editor.
6. You must also save in the object or editor window through which you accessed the VLDB Properties Editor. For example, if you accessed the VLDB properties by opening the Metric Editor and then opening the VLDB Properties Editor, after you click **Save and Close** in the VLDB Properties Editor, you must also click **Save and Close** in the Metric Editor to save your changes to VLDB properties.

Setting all VLDB properties to default

You can return all VLDB properties (those displayed in your chosen instance of the VLDB Properties Editor) to the default settings recommended for your database platform by MicroStrategy.

If you perform this procedure, any changes you may have made to any or all VLDB properties displayed in the chosen view of the VLDB Properties Editor will be lost. For details on which VLDB properties are displayed depending on how you access the VLDB Properties Editor, see *Details for all VLDB properties, page 27*.

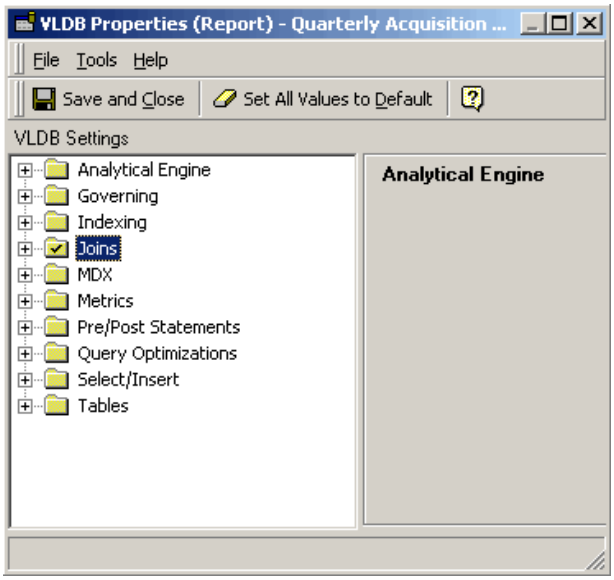
To set all VLDB property settings to their default status

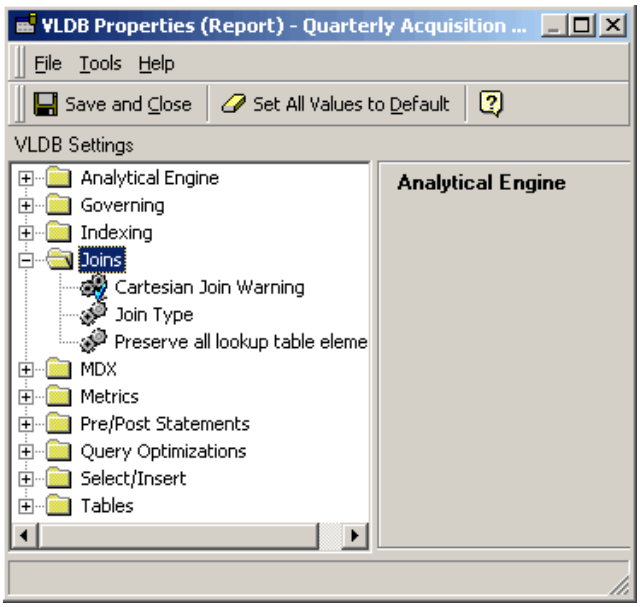
1. Use either or both of the following methods to see your system's VLDB properties that are not set to default. You should know which VLDB properties you will be affecting when you return properties to their default settings:
   - Generate a report listing VLDB properties that are not set to the default settings. For steps, see *Creating a VLDB settings report, page 20*, and select the check box named **Do not show settings with Default values**.

- Display an individual VLDB property by viewing the VLDB property whose default/non-default status you are interested in. (For steps, see *Viewing and changing VLDB properties, page 22*.) At the bottom of the Options and Parameters area for that property (on the right), you can see whether the **Use default inherited value** check box is selected. Next to this check box name, information appears about what level the setting is inheriting its default from.

2 Open the VLDB Properties Editor to display the VLDB properties that you want to set to their original defaults. (For information on object levels, see *Order of precedence, page 18*.)

3 In the VLDB Properties Editor, you can identify any VLDB properties that have had their default settings changed, because they are identified with a check mark. The folder in which the property is stored has a check mark on it (as shown on the Joins folder in the example image below), and the property name itself has a check mark on it (as shown on the gear icon in front of the Cartesian Join Warning property name in the second image below).





4 From the **Tools** menu, select **Set all values to default**. See the warning above if you are unsure about whether to set properties to the default.

5 In the confirmation window that appears, click **Yes**. All VLDB properties that are displayed in the VLDB Properties Editor are returned to their default settings.

6 Click **Save and Close** to save your changes and close the VLDB Properties Editor.

7 You must also save in the object or editor window through which you accessed the VLDB Properties Editor. For example, if you accessed the VLDB properties by opening the Metric Editor and then opening the VLDB Properties Editor, after you click **Save and Close** in the VLDB Properties Editor, you must also click **Save and Close** in the Metric Editor to save your changes to VLDB properties.

Upgrading the VLDB options for a particular database type

The database connection type specifies the type of database that the database instance represents, for example, Oracle 8i or Netezza 4.x. (The database connection type is specified on the General tab of the Database Instances Editor.) This setting ensures that the appropriate default VLDB properties, SQL syntax, and functions are used for your database type.

You must have Administrator privileges to upgrade the metadata. For information on upgrading the metadata and your MicroStrategy environment, see the *Upgrade Guide*. When the metadata updates the database type information:

- It loads new database types.
- It loads updated properties for existing database types that are still supported.
- It keeps properties for existing database types that are no longer supported. If an existing database type does not have any updates, but the properties for it have been removed, the process does not remove them from your metadata.

The steps below show you how to upgrade database types.

Prerequisites

- You have upgraded your MicroStrategy environment, as described in the *Upgrade Guide*.
- You have an account with administrative privileges.

To update database types

1. In Developer, log in to a project source using an account with administrative privileges.
2. From the **Folder List**, expand **Administration**, then **Configuration Managers**, and select **Database Instances**.
3. Right-click any database instance and select **Edit**. The Database Instances Editor opens.
4. To the right of the **Database connection type** drop-down list, click **Upgrade**. The Upgrade Database Type dialog box opens.
5. Click **Load** to load all the available database types for a MicroStrategy version.
6. Use the arrows to add any required database types by moving them from the **Available database types** list to the **Existing database types** list.
7. Click **OK** to save your changes and return to the Database Instances Editor.
8. Click **OK** to close the Database Instances Editor.

Modifying the VLDB properties for a warehouse database instance

If your database vendor updates its functionality, you may want to reset some VLDB properties in MicroStrategy. For example, if the timeout period is set too low and too many report queries are being cut off, you may want to modify the SQL Time Out (Per Pass) setting.

For descriptions and examples of all VLDB properties and to see what properties can be modified, see *Details for all VLDB properties, page 27*.

To modify the VLDB properties related to a database instance, use the appropriate steps from the table in *Opening the VLDB Properties Editor, page 19* to access the VLDB Properties Editor for the database instance. Then follow the steps for *Viewing and changing VLDB properties, page 22*.

Details for all VLDB properties

Modifying any VLDB property should be performed with caution only after understanding the effects of the VLDB settings you want to apply. A given VLDB setting can support or optimize one system setup, but the same setting can cause performance issues or errors for other systems. Use this manual to learn about the VLDB properties before modifying any default settings.

Subtotals over Consolidations Compatibility

Consolidations allow users to group specific attribute elements together and place the group on a report template as if the group was an attribute. The elements of a consolidation can have arithmetic calculations performed on them. The Subtotals over Consolidations Compatibility property allows you to determine how the Analytical Engine calculates consolidations.

- **Evaluate subtotals over consolidation elements and their corresponding attribute elements (behavior for 7.2.x and earlier)** (default): In MicroStrategy version 7.2.x and earlier, if a calculation includes a consolidation, the Analytical Engine calculates subtotals across the consolidation elements as well as across all attribute elements that comprise the consolidation element expressions.
- **Evaluate subtotals over consolidation elements only (behavior for 7.5 and later)**: In MicroStrategy version 7.5 and later, if a calculation includes a consolidation this setting allows the Analytical Engine to calculate only those elements that are part of the consolidation.

When you enable this setting, be aware of the following requirements and options:

This VLDB property must be set at the project level for the calculation to be performed correctly.

The setting takes effect when the project is initialized, so after this setting is changed you must reload the project or restart Intelligence Server.

After you enable this setting, you must enable subtotals at either the consolidation level or the report level. If you enable subtotals at the consolidation level, subtotals are available for all reports in which the consolidation is used. (Consolidation Editor -> Elements menu -> Subtotals -> Enabled.) If you enable subtotals at the report level, subtotals for consolidations can be enabled on a report-by-report basis. (Report Editor -> Report Data Options -> Subtotals -> Yes. If Default is selected, the Analytical Engine reverts to the Enabled/Disabled property as set on the consolidation object itself.)

If the project is registered on an Intelligence Server version 7.5.x but is accessed by clients using Developer version 7.2.x or earlier, leave this property setting on "Evaluate subtotals over consolidation elements and their corresponding attribute elements." Otherwise, metric values may return as zeroes when Developer 7.2.x users execute reports with consolidations, or when they pivot in such reports.

Change this property from the default only when all Developer clients have upgraded to MicroStrategy version 7.5.x.

Levels at which you can set this

Project only

Example

Three consolidations called Super Regions are created, defined as follows:

- East (({Cust Region=Northeast} + {Cust Region=Mid-Atlantic}) + {Cust Region=Southeast})
- Central ({Cust Region=Central} + {Cust Region=South})
- West ({Cust Region=Northwest} + {Cust Region=Southwest})

With the first setting selected, "Evaluate subtotals over consolidation elements and their corresponding attribute elements," the report appears as follows:

Super Regions	Metrics	Dollar Sales
East		7,252,458
Central		5,574,740
West		4,109,236
Total		33,872,868

The Total value is calculated for more elements than are displayed in the Super Regions column. The Analytical Engine is including the following elements in the calculation: East + (Northeast + Mid-Atlantic + Southeast) + Central + (Central + South) + West + (Northwest + Southwest).

With the second setting selected, "Evaluate subtotals over consolidation elements only," and with subtotals enabled, the report appears as follows:

Super Regions	Metrics	Dollar Sales
East		7,252,458
Central		5,574,740
West		4,109,236
Total		16,936,434

The Total value is now calculated for only the Super Regions consolidation elements. The Analytical Engine is including only the following elements in the calculation: East + Central + West.

Apply Filter Options for queries against in-memory datasets

Apply Filter Options for queries against in-memory datasets is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Apply Filter Options for queries against in-memory datasets determines how many times the view filter is applied, which can affect the final view of data. For example, consider the simple report shown below, which shows yearly cost:

Year	Metrics	Cost
2012		$7,343,097
2013		$9,777,521
2014		$12,609,467

You create a Yearly Cost derived metric that uses the following definition:

```
Sum(Cost){!Year%}
```

The level definition of `{!Year%}` defines the derived metric to ignore filtering related to Year and to perform no grouping related to Year (for explanation and examples of defining the level for metrics, see the *Advanced Reporting Guide*). This means that this derived metric displays the total cost for all years, as shown in the report below:

Year	Metrics	Yearly Cost	Cost
2012		$29,730,085	$7,343,097
2013		$29,730,085	$9,777,521
2014		$29,730,085	$12,609,467

You can also further filter this report using a view filter. For example, a view filter is applied to this report, which restricts the results to only 2014, as shown below:

Year	Metrics	Yearly Cost	Cost
2014		$29,730,085	$12,609,467

By default, only Cost for 2014 is displayed, but Yearly Cost remains the same since it has been defined to ignore filtering and grouping related to Year. This is supported by the default option Apply view filter to passes touching fact tables and last join pass of the Apply Filter Options for queries against in-memory datasets VLDB property.

If analysts of this report are meant to be more aware of the cost data that goes into the total of Yearly Cost, you can modify the Apply Filter Options for queries against in-memory datasets VLDB property to use the option Apply view filter only to passes touching fact tables. This displays the other elements of Year, as shown in the report below:

Year	Metrics	Yearly Cost	Cost
2012		$29,730,085	
2013		$29,730,085	
2014		$29,730,085	$12,609,467

You have the following options for the Apply Filter Options for queries against in-memory datasets VLDB property:

- **Apply view filter only to passes touching fact tables**: This option applies the view filter to only SQL passes that touch fact tables, but not to the last pass that combines the data. As shown in the example above, this can include additional information on the final display by removing the view filter from the final display of the report.
- **Apply view filter to passes touching fact tables and last join pass** (default): This option applies the view filter to SQL passes that touch fact tables as well as the last pass that combines the data. As shown in the example above, this applies the view filter to the final display of the report to ensure that the data meets the restrictions defined by the view filter.

Levels at which you can set this

Project, report, and template

Custom Group Display for Joint Elements

The Custom Group Display for Joint Elements VLDB property determines whether to display all attribute elements or just a single attribute element for custom groups that include multiple attributes for a single custom group element. A custom group must meet the following criteria for this VLDB property to affect the display of the custom group elements:

- Two or more attributes are included in the qualifications for a single custom group element. This includes custom group elements that are defined using the following filtering techniques:
  - Multiple filter qualifications that are based on attributes are used to define a custom group element. For example, you can include one filter qualification that filters data for only the year 2011, and another filter qualification that filters data for the Northeast region. This would include both the attributes Year and Region for the custom group element. Steps to create filter qualifications for custom group elements are provided in the *Advanced Reporting Guide*.
  - A joint element list is used to define the custom group element. A joint element list is a filter that allows you to join attribute elements and then filter on that attribute result set. In other words, you can select specific element combinations, such as quarter and category. Steps to create a joint element list are provided in the *Advanced Reporting Guide*.
- The individual attribute elements must be displayed for each custom group element. For steps to display the individual attribute elements for a custom group element, see the *Advanced Reporting Guide*.

For custom groups that meet the criteria listed above, the Custom Group Display for Joint Elements VLDB property provides the following formatting options:

- **Display element names from all attributes in the joint element** (default): Displays all of the attribute elements that are included in the filter qualifications for the custom group element. For example, the attributes Region and Category are used in a joint element list, which is then used to create a custom group element. When this custom group is included in a report, the attribute elements, for each qualification of the joint element list, are displayed for the custom group elements, as shown in the report below.

Example Custom Group	Metrics	Revenue
Region and Category Sales		$4,152,484
Books | Northeast		$646,421
Electronics | Central		$3,506,062

  The attribute elements for both Region and Category are displayed for each custom group element.

- **Display element names from only the first attribute in the joint element**: Displays only one attribute element for the attributes that are included in the filter qualifications for the custom group element. An attribute element from the attribute that is first in terms of alphabetical order is displayed for the custom group. For example, the attributes Region and Category are used in separate filter qualifications, which are then used to create a custom group element. When this custom group is included in a report, the Category attribute element is displayed for the custom group elements, as shown in the report below.

Example Custom Group	Metrics	Revenue
Region and Category Sales		$4,152,484
Books		$646,421
Electronics		$3,506,062

  Only the attribute elements for the Category attribute are displayed. The attribute elements for Region are not displayed because Category is first in terms of alphabetical order.

Levels at which you can set this

Project only

Display NULL On Top

The Display NULL on Top property determines where NULL values appear when you sort data. The default is to display the NULL values at the top of a list of values when sorting.

Wherever NULL values occur in a report, they are displayed as user-defined strings. NULL values result from a variety of scenarios. NULL values can come from data retrieved from the database, from cross-tabulation on a report, or from data aggregation

on a report. You can determine what characters or strings are displayed for NULL values. To do this, access the Project Configuration Editor, select the **Report definition: Null values** category, then type the strings you want to have displayed, in the appropriate fields.

Levels at which you can set this

Project, report, and template

Document Grids from Multiple Datasets

The Document Grids from Multiple Datasets property determines whether objects in Grid/Graphs in documents must come from a single dataset or can come from multiple datasets.

- **Objects in document grids must come from the grid's source dataset**: If you select this option, objects in a Grid/Graph must come from a single dataset, the source dataset used by the Grid/Graph. For example, a document contains two datasets. Dataset 1 contains Region and Revenue; Dataset 2 contains Region and Profit. You cannot create a Grid/Graph with Region, Revenue, and Profit. You can use this option for backwards compatibility with existing documents.
- **Allow objects in document grids to come from multiple datasets**: By default, a single Grid/Graph can contain objects from multiple datasets, providing additional levels of data analysis. A Grid/Graph can contain Region and Revenue from Dataset 1 as well as Profit from Dataset 2.

For background information on creating Grid/Graphs in documents, including using multiple datasets on a single Grid/Graph, see the *Report Services Document Creation Guide*.

Levels at which you can set this

Project only

Evaluation Ordering

Evaluation Ordering is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

An evaluation order is the order in which the MicroStrategy Analytical Engine performs different kinds of calculations during the data population stage. The Evaluation Ordering property determines the order in which calculations are resolved. MicroStrategy objects that are included in the evaluation order include consolidations, compound smart metrics, report limits, subtotals, derived metrics, and derived elements. Some result data can differ depending on the evaluation order of these objects.

- **6.x order - Calculate derived metric/smart compound metric before derived elements/consolidation and all subtotals as smart**: This option is

used primarily to support backward compatibility. It is recommended in most scenarios to update your project to use the 9.x evaluation order described below.

- **7.x order - Calculate derived metric/smart compound metric before derived elements/consolidation and all subtotals as non-smart**: This option allows you to modify the order of certain calculations relative to the default 9.x order. Additionally, all subtotals including the total subtotal are not calculated as smart subtotals. Smart subtotals are commonly used to calculate subtotals that provide ratios or percentages.
- **9.x order - Calculate derived elements/consolidation before derived metric/smart compound metric, "total" subtotal as smart and other subtotals as non-smart**: This default option is recommended in most scenarios. For example, calculating the total subtotal as a smart subtotal allows it to calculate ratios and percentages accurately in most cases. The order of the other calculations also supports the most common data analysis requirements. A common case that can require a different evaluation order than the default 9.x order is the calculation and display of ratio and percentage values. If your report does not display values as expected, select the other evaluation orders for your report and re-execute the report to view the new results.

To review the evaluation order of a report, in Developer, view the report in SQL View. In the SQL View, the section listed as `Analytical engine calculation steps` describes the order in which the various report objects are evaluated. To change the evaluation order for a report using Developer, on the Report Editor, from the **Data** menu, select **Report Data Options**. The Report Data Options dialog box opens. Expand the **Calculations** category, and select **Evaluation Order**. Clear the **Use default evaluationorder** check box to define your own evaluation order.

For examples of how you can modify the evaluation order of objects on a project, see the *Advanced Reporting Guide*.

Levels at which you can set this

Project, report, and template

Filtering on String Values

The Filtering on String Values VLDB property determines whether filters consider trailing spaces in attribute elements. This can affect the data that is restricted when filtering data. This VLDB property has the following options:

- **Do not trim trailing spaces**: Attribute elements that include trailing spaces can be returned as separate attribute elements when filtering on the attribute. For example, an attribute has two attribute elements, one with the description information "South" and the other with the description information "South " which has an extra trailing space at the end. By selecting this option, these attribute elements can be returned as separate attribute elements when filtering on the attribute.
- **Trim trailing spaces**: Attribute elements that include trailing spaces are not returned as separate attribute elements when filtering on the attribute. Instead, any trailing spaces are ignored. For example, an attribute has two attribute elements, one

with the description information “South” and the other with the description information “South ” which has an extra trailing space at the end. By selecting this option, only a single South attribute element is returned when filtering on the attribute. Since trailing spaces are commonly viewed as an error in the data, it is recommended that you use this default Trim trailing spaces option to ignore any trailing spaces.

Levels at which you can set this

Project only

Metric Level Determination

The Metric Level Determination VLDB property defines how the level of data is stored for reports. This level is used to make other determinations for the report such as the level at which to report metric data. This VLDB property has the following options:

- **Include only lowest-level attributes in metric level** (default): The report’s level is defined using only the lowest-level attributes available in the report. This option correctly reflects the level of data in the report while also optimizing the amount of resources required to define the level of the report.

  For example, a report includes the attributes Year, Month, Category, and Subcategory. The Year and Month attributes are from the same hierarchy and Month is the lowest-level attribute from that hierarchy on the report. Similarly, the Category and Subcategory attributes are from the same hierarchy and Subcategory is the lowest-level attribute from that hierarchy on the report. When selecting this option for the Metric Level Determination VLDB property, the level of the report is defined as Month and Subcategory. It can be defined in this way because these are the lowest-level attributes from the hierarchies that are present on the report.

  This level can then be used with metrics to determine the level at which their data must be reported. If the physical schema of your project matches the expected logical schema, correct metric data is displayed and the resources required to determine the report level are optimized.

- **Include higher-level related attributes in metric level**: The report’s level is defined using all attributes available in the report. This option correctly reflects the level of data in the report, but it can require additional resources to define the level of the report.

  Consider the example used to describe the previous option. If the physical schema of your project matches the expected logical schema, then including only the lowest-level attributes displays correct metric data. However, differences between your physical schema and expected logical schema can cause unexpected data to be displayed if only the lowest level attributes are used to define the level of the report.

  For example, while the relationship between the Category and Subcategory attributes is defined as a one-to-many relationship, the data in your data source reflects a many-to-many relationship. Because of this mismatch, including only the lowest-level attributes can return unexpected metric data. By selecting this option for the Metric Level Determination VLDB property, the additional higher-level attributes are included when defining the level of the report and can be used to return the metric

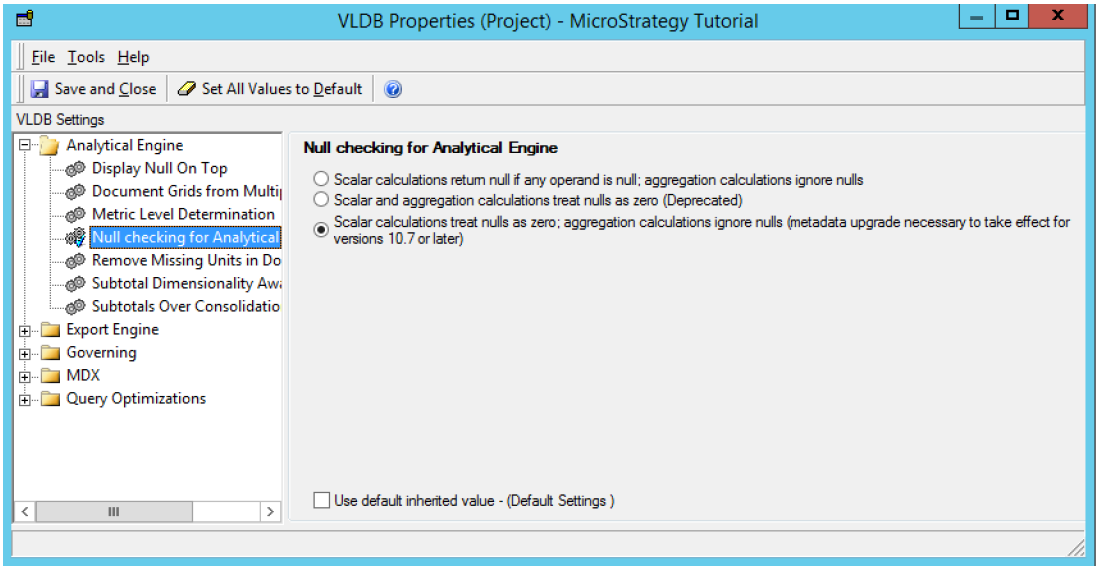
data as it exists in the data source. However, while this helps return accurate data in these types of scenarios, the higher-level attributes require additional resources to define the level of the report.

Levels at which you can set this

Project and report

NULL checking for Analytical Engine

The Null checking for Analytical Engine property determines whether or not null values are interpreted as zeros when the Analytical Engine calculates data.



The default option is for aggregation calculations to ignore nulls and for scalar calculations to treat null values as zero. Any projects that existed prior to upgrading metadata to MicroStrategy 10.8 will retain their original VLDB property settings. For more information on this setting, see the Advanced Reporting Guide.

Changes made to this VLDB setting can cause differences to appear in your data output. Metrics using count or average, metrics with dynamic aggregation set to count or average, as well as thresholds based on such metrics could be impacted by altered calculation behavior.

Levels at which you can set this

Project, report, template, and metric

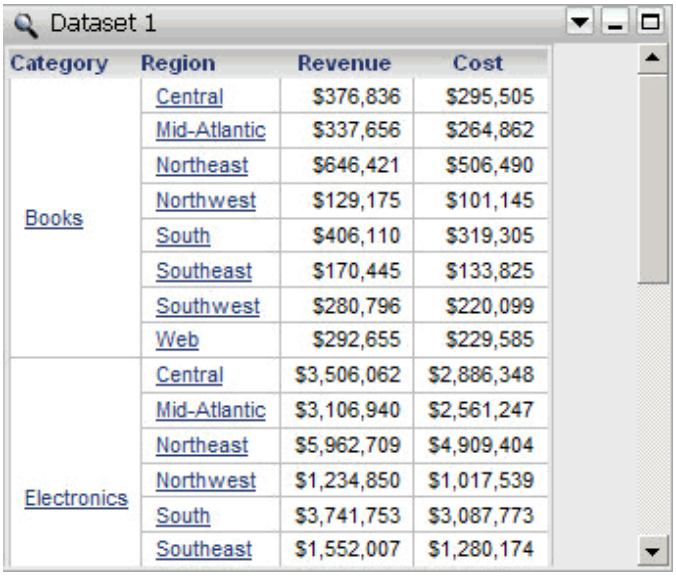
Remove Missing Units in Documents

If you remove or replace a dataset, controls on the document that contain data that is no longer available from the dataset will be updated and will no longer contain data from

the replaced or removed dataset. For a Grid/Graph, objects that are available in another dataset are updated to contain data from the other dataset. The Remove Missing Units in Documents property then determines how any objects that are still missing are displayed in Grid/Graphs:

- **Remove objects not available in the source dataset(s)**: The missing objects are not displayed in the Grid/Graph. If the Grid/Graph only contains missing objects, it is displayed as an empty placeholder.
- **Do not remove objects not available in the source dataset(s)**: The headers for the missing objects are displayed in the Grid/Graph, without any data. MicroStrategy recommends that objects missing from datasets are displayed. This can alert you if objects are removed from a report used as a dataset.

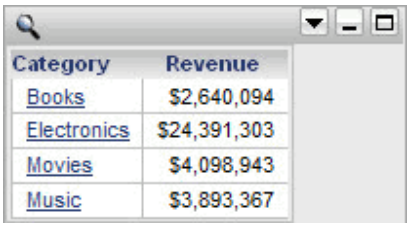
For example, a document contains two datasets. Dataset 1 has Category, Region, and the Revenue and Cost metrics. Dataset 2 has Category, Subcategory, and the Revenue and Profit metrics. A Grid/Graph containing the objects from Dataset 1 is displayed on the document. A portion of the Grid/Graph is shown below, in Interactive Mode in MicroStrategy Web:



Category	Region	Revenue	Cost
Books	Central	$376,836	$295,505
	Mid-Atlantic	$337,656	$264,862
	Northeast	$646,421	$506,490
	Northwest	$129,175	$101,145
	South	$406,110	$319,305
	Southeast	$170,445	$133,825
	Southwest	$280,796	$220,099
	Web	$292,655	$229,585
Electronics	Central	$3,506,062	$2,886,348
	Mid-Atlantic	$3,106,940	$2,561,247
	Northeast	$5,962,709	$4,909,404
	Northwest	$1,234,850	$1,017,539
	South	$3,741,753	$3,087,773
	Southeast	$1,552,007	$1,280,174

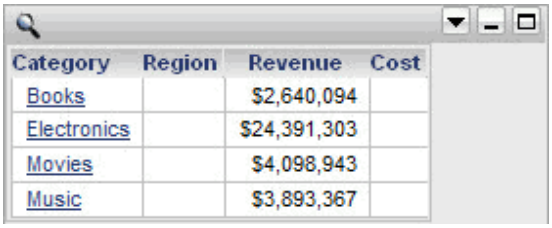
Dataset 1 is removed from the document. Because Category and Revenue are available from Dataset 2, they continue to be displayed on the Grid/Graph. Since Region and Cost are no longer available in any dataset on the document, they are considered missing objects. Which option is selected in the Remove Missing Units in Documents property then determines how any objects that are still missing are displayed in Grid/Graphs, as described below:

- **Remove objects not available in the source dataset(s)**: The missing objects are not displayed in the Grid/Graph, as shown below:



Category	Revenue
Books	$2,640,094
Electronics	$24,391,303
Movies	$4,098,943
Music	$3,893,367

- **Do not remove objects not available in the source dataset(s)**: Headers for the missing objects are displayed in the Grid/Graph, as shown below:



Category	Region	Revenue	Cost
Books		$2,640,094	
Electronics		$24,391,303	
Movies		$4,098,943	
Music		$3,893,367	

Regardless of the property setting, a text field that contains a dataset object (such as an attribute or a metric) will display the object name instead of values. For example, a text field displays {Region} instead of North, South, and so on.

For an example that uses multiple datasets in a single Grid/Graph, see the Document Creation Guide.

Levels at which you can set this

Project and document

Subtotal Dimensionality Aware

MicroStrategy 7i (7.2.x and later) has the ability to detect the level of a metric and subtotal it accordingly. The Subtotal Dimensionality Aware property allows you to choose between the 7.1 and earlier subtotaling behavior (FALSE) and the 7.2.x and later subtotaling behavior (TRUE). MicroStrategy recommends that you set this property to TRUE.

If this property is set to True, and a report contains a metric that is calculated at a higher level than the report level, the subtotal of the metric is calculated based on the metric's level. For example, a report at the Quarter level containing a yearly sales metric shows the yearly sales as the subtotal instead of simply summing the rows on the report.

Levels at which you can set this

Project, report, template, and metric

Example

Quarterly Dollar Sales metric is defined as

```
Sum(Revenue) Dimensionality = Quarter
```

Yearly Dollar Sales metric is defined as

```
Sum(Revenue) Dimensionality = Year
```

Year	Quarter	Quarterly Dollar Sales	Yearly Dollar Sales
2002	1	100	600
2002	2	200	600
2002	3	100	600
2002	4	200	600
	Grand Total	600	**2400** or **600** depending on the setting (see below)

If Subtotal Dimensionality Aware is set to FALSE

The quarterly subtotal is calculated as 600, that is, a total of the Quarterly Dollar Sales values. The yearly subtotal is calculated as 2400, the total of the Yearly Dollar Sales values. This is how MicroStrategy 7.1 calculates the subtotal.

If Subtotal Dimensionality Aware is set to TRUE

The quarterly subtotal is still 600. Intelligence Server is aware of the level of the Yearly Dollar Sales metric, so rather than simply adding the column values, it correctly calculates the Yearly Dollar Sales total as 600.

Providing access to Intelligent Cube data: Dynamic sourcing

The table below summarizes the Dynamic sourcing VLDB properties. Additional details about each property, including examples where necessary, are available by clicking on the links in the table.

Property	Description	Possible Values	Default Value
Aggregate Table Validation	Defines whether dynamic sourcing is enabled or disabled for aggregate tables.	• Aggregate tables contain the same data as corresponding detail tables and the aggregation function is SUM • Aggregate tables contain either less data or more data than their corresponding detail tables and/or the aggregation function is not SUM	Aggregate tables contain the same data as corresponding detail tables and the aggregation function is SUM

Property	Description	Possible Values	Default Value
Attribute Validation	Defines whether dynamic sourcing is enabled or disabled for attributes.	• Attribute columns in fact tables and lookup tables do not contain NULLs and all attribute elements in fact tables are present in lookup tables • Attribute columns in fact tables or lookup tables may contain NULLs and/or some attribute elements in fact tables are not present in lookup tables	Attribute columns in fact tables and lookup tables do not contain NULLs and all attribute elements in fact tables are present in lookup tables
Enable Cube Parse Log in SQL View	Defines whether the Intelligent Cube Parse log is displayed in the SQL View of an Intelligent Cube. This log helps determine which reports use dynamic sourcing to connect to the Intelligent Cube.	• Disable Cube Parse Log in SQL View • Enable Cube Parse Log in SQL View	Disable Cube Parse Log in SQL View
Enable Dynamic Sourcing for Report	Defines whether dynamic sourcing is enabled or disabled for reports.	• Disable dynamic sourcing for report • Enable dynamic sourcing for report	Enable dynamic sourcing for report
Enable Extended Mismatch Log in SQL View	Defines whether the extended mismatch log is displayed in the SQL View of a report. The extended mismatch log helps determine why a metric prevents the use of dynamic sourcing is provided in the extended mismatch log.	• Disable Extended Mismatch Log in SQL View • Enable Extended Mismatch Log in SQL View	Disable Extended Mismatch Log in SQL View
Enable Mismatch Log in SQL View	Defines whether the mismatch log is displayed in the SQL View of a report. This log helps determine why a report that can use dynamic sourcing cannot connect to a specific Intelligent Cube.	• Disable Mismatch Log in SQL View • Enable Mismatch Log in SQL View	Disable Mismatch Log in SQL View
Enable Report Parse Log in SQL View	Defines whether the Report Parse log is displayed in the SQL View of a report. This log helps determine whether the report can use dynamic sourcing to	• Disable Report Parse Log in SQL View • Enable Report Parse Log in SQL View	Disable Report Parse Log in SQL View

Property	Description	Possible Values	Default Value
	connect to an Intelligent Cube.		
Metric Validation	Defines whether dynamic sourcing is enabled or disabled for metrics.	• Enable dynamic sourcing for metric • Disable dynamic sourcing for metric	Enable dynamic sourcing for metric
String Comparison Behavior	Defines whether dynamic sourcing is enabled or disabled for attributes that are used in filter qualifications.	• Use case insensitive string comparison with dynamic sourcing • Do not allow any string comparison with dynamic sourcing	Use case insensitive string comparison with dynamic sourcing

Aggregate Table Validation

Aggregate Table Validation is an advanced VLDB property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Reports that use aggregate tables are available for dynamic sourcing by default, but there are some data modeling conventions that should be considered when using dynamic sourcing.

In general, aggregate tables allow accurate data to be returned to reports from Intelligent Cubes through dynamic sourcing. However, if the aggregate tables use an aggregation other than Sum, or there is different data between aggregate tables and other tables in the data warehouse, this can cause aggregate tables to return incorrect data when dynamic sourcing is used. An example of an aggregate table not containing the same data is if an aggregate table includes data for years 2006, 2007, and 2008 but the lookup table for Year only includes data for 2007 and 2008.

You can enable and disable dynamic sourcing for aggregate tables by modifying the Aggregate Table Validation VLDB property. This VLDB property has the following options:

- **Aggregate tables contain the same data as corresponding detail tables and the aggregation function is SUM** (default): This is the default option for aggregate tables, which enables aggregate tables for dynamic sourcing.
- **Aggregate tables contain either less data or more data than their corresponding detail tables and/or the aggregation function is not SUM**: This option disables dynamic sourcing for aggregate tables. This setting should be used if your aggregate tables are not modeled to support dynamic sourcing. The use of an aggregation function other than Sum or the mismatch of data in your aggregate tables with the rest of your data warehouse can cause incorrect data to be returned to reports from Intelligent Cubes through dynamic sourcing.

You can disable dynamic sourcing individually for reports that use aggregate tables or you can disable dynamic sourcing for all reports that use aggregate tables within a project.

While the definition of the VLDB property at the project level defines a default for all reports in the project, any modifications at the report level take precedence over the project level definition. For information on defining a project-wide dynamic sourcing strategy, see the *In-memory Analytics Guide*.

Levels at which you can set this

Project, report, and template

Attribute Validation

Attribute Validation is an advanced VLDB property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Attributes are available for dynamic sourcing by default, but there are some data modeling conventions that should be considered when using dynamic sourcing.

In general, if attributes use outer joins accurate data can be returned to reports from Intelligent Cubes through dynamic sourcing. However, if attributes use inner joins, which is a more common join type, you should verify that the attribute data can be correctly represented through dynamic sourcing.

Two scenarios can cause attributes that use inner joins to return incorrect data when dynamic sourcing is used:

- Attribute information in lookup and fact tables includes NULL values.
- All attribute elements in fact tables are not also present in lookup tables.

You can enable and disable dynamic sourcing for attributes by modifying the Attribute Validation VLDB property. This VLDB property has the following options:

- **Attribute columns in fact tables and lookup tables do not contain NULLs and all attribute elements in fact tables are present in lookup tables** (default): This option enables attributes for dynamic sourcing.
- **Attribute columns in fact tables and lookup tables may contain NULLs and/or some attribute elements in fact tables are not present in lookup tables**: This option disables dynamic sourcing for attributes. This setting should be used if your attribute data is not modeled to support dynamic sourcing. The inclusion of NULLs in your attribute data or a mismatch between available attribute data in your fact and lookup tables can cause incorrect data to be returned to reports from Intelligent Cubes through dynamic sourcing.

You can disable dynamic sourcing for attributes individually or you can disable dynamic sourcing for all attributes within a project. While the definition of the VLDB property at the project level defines a default for all attributes in the project, any modifications at the attribute level take precedence over the project level definition. For information on defining a project-wide dynamic sourcing strategy, see the In-memory Analytics Guide.

Levels at which you can set this

Project and attribute

Enable Cube Parse Log in SQL View

Enable Cube Parse Log in SQL View is an advanced VLDB property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Intelligent Cube parse log helps determine which reports use dynamic sourcing to connect to an Intelligent Cube, as well as why some reports cannot use dynamic sourcing to connect to an Intelligent Cube. By default, the Intelligent Cube parse log can only be viewed using the MicroStrategy Diagnostics and Performance Logging tool. You can also allow this log to be viewed in the SQL View of an Intelligent Cube.

This VLDB property has the following options:

- **Disable Cube Parse Log in SQL View** (default): This option allows the Intelligent Cube parse log to only be viewed using the MicroStrategy Diagnostics and Performance Logging tool.
- **Enable Cube Parse Log in SQL View**: Select this option to allow the Intelligent Cube parse log to be viewed in the SQL View of an Intelligent Cube. This information can help determine which reports use dynamic sourcing to connect to the Intelligent Cube.

Levels at which you can set this

- Intelligent Cube and project

Enable Dynamic Sourcing for Report

Enable Dynamic Sourcing for Report is an advanced VLDB property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

By default, dynamic sourcing is disabled for reports, and they therefore retrieve their results by running against the data warehouse. You can enable dynamic sourcing for a report so that active Intelligent Cubes (that are also enabled for dynamic sourcing) are checked to see if the report can retrieve its data from an Intelligent Cube. If an Intelligent Cube fits the data requirements of a report, the report can be run without executing against the data warehouse.

You can enable dynamic sourcing for reports by modifying the Enable Dynamic Sourcing for Report VLDB property. This VLDB property has the following options:

- **Disable dynamic sourcing for report**: Dynamic sourcing is disabled for reports.
- **Enable dynamic sourcing for report** (default): Dynamic sourcing is enabled for reports.

You can enable dynamic sourcing for reports individually or you can enable dynamic sourcing for all reports within a project. While the definition of the VLDB property at the project level defines a default for all reports in the project, any modifications at the report level take precedence over the project level definition. For information on defining a project-wide dynamic sourcing strategy, see the *In-memory Analytics Guide*.

Levels at which you can set this

Project, report, and template

Enable Extended Mismatch Log in SQL View

Enable Extended Mismatch Log in SQL View is an advanced VLDB property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The extended mismatch log helps determine why a metric prevents the use of dynamic sourcing is provided in the extended mismatch log. This information is listed for every metric that prevents the use of dynamic sourcing. By default, the extended mismatch log can only be viewed using the MicroStrategy Diagnostics and Performance Logging tool. You can also allow this log to be viewed in the SQL View of a report.

The extended mismatch log can increase in size quickly and thus is best suited for troubleshooting purposes.

This VLDB property has the following options:

- **Disable Extended Mismatch Log in SQL View** (default): This option allows the extended mismatch log to only be viewed using the MicroStrategy Diagnostics and Performance Logging tool.
- **Enable Extended Mismatch Log in SQL View**: Select this option to allow the extended mismatch log to be viewed in the SQL View of a report. This information can help determine why a report that can use dynamic sourcing cannot connect to a specific Intelligent Cube.

Levels at which you can set this

Report, template, and project

Enable Mismatch Log in SQL View

Enable Mismatch Log in SQL View is an advanced VLDB property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The mismatch log helps determine why a report that can use dynamic sourcing cannot connect to a specific Intelligent Cube. By default, the mismatch log can only be viewed using the MicroStrategy Diagnostics and Performance Logging tool. You can also allow this log to be viewed in the SQL View of a report.

This VLDB property has the following options:

- **Disable Mismatch Log in SQL View** (default): This option allows the mismatch log to only be viewed using the MicroStrategy Diagnostics and Performance Logging tool.

- **Enable Mismatch Log in SQL View**: Select this option to allow the mismatch log to be viewed in the SQL View of a report. This information can help determine why a report that can use dynamic sourcing cannot connect to a specific Intelligent Cube.

Levels at which you can set this

Report, template, and project

Enable Report Parse Log in SQL View

Enable Report Parse Log in SQL View is an advanced VLDB property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The report parse log helps determine whether the report can use dynamic sourcing to connect to an Intelligent Cube. By default, the report parse log can only be viewed using the MicroStrategy Diagnostics and Performance Logging tool. You can also allow this log to be viewed in the SQL View of a report.

This VLDB property has the following options:

- **Disable Report Parse Log in SQL View** (default): This option allows the report parse log to only be viewed using the MicroStrategy Diagnostics and Performance Logging tool.
- **Enable Report Parse Log in SQL View**: Select this option to allow the report parse log to be viewed in the SQL View of a report. This information can help determine whether the report can use dynamic sourcing to connect to an Intelligent Cube.

Levels at which you can set this

Report, template, and project

Metric Validation

Metric Validation is an advanced VLDB property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Metrics are available for dynamic sourcing by default, but there are some data modeling conventions that should be considered when using dynamic sourcing.

In general, if metrics use outer joins, accurate data can be returned to reports from Intelligent Cubes through dynamic sourcing. However, if metrics use inner joins, which is a more common join type, you should verify that the metric data can be correctly represented through dynamic sourcing.

If the fact table that stores data for metrics includes NULL values for metric data, this can cause metrics that use inner joins to return incorrect data when dynamic sourcing is used. This scenario is uncommon.

You can enable and disable dynamic sourcing for metrics by modifying the Metric Validation VLDB property. This VLDB property has the following options:

- **Enable dynamic sourcing for metric** (default): This option enables metrics for dynamic sourcing.
- **Disable dynamic sourcing for metric**: This option disables dynamic sourcing for metrics. This setting should be used if your metric data is not modeled to support dynamic sourcing. The inclusion of NULLs in fact tables that contain your metric data can cause incorrect data to be returned to reports from Intelligent Cubes through dynamic sourcing.

You can disable dynamic sourcing for metrics individually or you can disable dynamic sourcing for all metrics within a project. While the definition of the VLDB property at the project level defines a default for all metrics in the project, any modifications at the metric level take precedence over the project level definition. For information on defining a project-wide dynamic sourcing strategy, see the *In-memory Analytics Guide*.

Levels at which you can set this

Project and metric

String Comparison Behavior

String Comparison Behavior is an advanced VLDB property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

To ensure that dynamic sourcing can return the correct results for attributes, you must also verify that filtering on attributes achieves the same results when executed against your database versus an Intelligent Cube.

The results returned from a filter on attributes can potentially return different results when executing against the database versus using dynamic sourcing to execute against an Intelligent Cube. This can occur if your database is case-sensitive and you create filter qualifications that qualify on the text data of attribute forms.

If your database is case-sensitive, this is enforced for the filter qualification. However, filtering for an Intelligent Cube is handled by the Analytical Engine which does not enforce case sensitivity.

Consider a filter qualification that filters on customers that have a last name beginning with the letter h. If your database is case-sensitive and uses uppercase letters for the first letter in a name, a filter qualification using a lowercase h is likely to return no data. However, this same filter qualification on the same data stored in an Intelligent Cube returns all customers that have a last name beginning with the letter h.

You can configure this dynamic sourcing behavior for attributes by modifying the String Comparison Behavior VLDB property. This VLDB property has the following options:

- **Use case insensitive string comparison with dynamic sourcing** (default): When attempting to use dynamic sourcing, it allows filter qualifications to qualify on the text data of attribute forms without enforcing case sensitivity.

This is a good option if your database does not enforce case sensitivity. In this scenario, dynamic sourcing returns the same results that would be returned by the filter qualification if the report was executed against the database.

- **Do not allow any string comparison with dynamic sourcing**: This option disables dynamic sourcing for attributes when a filter qualification is used to qualify on the text data of attribute forms.

  This is a good option if your database is case sensitive. In this scenario, dynamic sourcing could return different results than what would be returned by the filter qualification if the report was executed against the database.

You can modify this VLDB property for attributes individually or you can modify it for all attributes within a project. While the definition of the VLDB property at the project level defines a default for all attributes in the project, any modifications at the attribute level take precedence over the project level definition. For information on defining a project-wide dynamic sourcing strategy, see the *In-memory Analytics Guide*.

Levels at which you can set this

Project and attribute

Exporting report results from MicroStrategy: Export Engine

The table below summarizes the Export Engine VLDB properties. Additional details about each property, including examples where necessary, are provided in the sections following the table.

Property	Description	Possible Values	Default Value
GUID of attributes in profit and loss hierarchy (separated by ':') that has dummy rows to be removed	Lets you identify attributes that include empty elements, which can then be ignored when exporting to Microsoft Excel or to a PDF file.	A list of attribute ID values, each one separated using a colon (:).	NULL

GUID of attributes in profit and loss hierarchy (separated by ':') that has dummy rows to be removed

GUID of attributes in profit and loss hierarchy (separated by ':') that has dummy rows to be removed is an advanced property that is hidden by default. For instructions on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The GUID of attributes in profit and loss hierarchy (separated by ':') that has dummy rows to be removed VLDB property lets you identify attributes that include empty elements, which can then be ignored when exporting to Microsoft Excel or to a PDF file. This is useful when creating financial line item attributes as part of supporting a financial reporting solution in MicroStrategy. For a detailed explanation of how to support

financial reporting in MicroStrategy, along with using this VLDB property to identify attributes that include empty elements, refer to the *Project Design Guide*.

To identify attributes that include empty elements, type the ID value for each attribute in the text field for this VLDB property. To determine the ID value for an attribute object, navigate to an attribute in Developer, right-click the attribute, and then select Properties. Details about the attribute, including the ID value are displayed.

If you need to identify multiple attributes as having empty elements, separate each attribute ID using a colon (:).

Levels at which you can set this

Project only

Customizing SQL queries: Freeform SQL

The table below summarizes the Freeform SQL VLDB properties. Additional details about each property, including examples where necessary, are provided in the sections following the table.

Property	Description	Possible Values	Default Value
Ignore Empty Result for Freeform SQL	Provides the flexibility to display or hide warnings when a Freeform SQL statement returns an empty result.	• Do not turn off warnings for Freeform SQL statements with empty results, such as updates. • Turn off warnings for Freeform SQL statements with empty results, such as updates. • Turn off warnings for Freeform SQL statements that return multiple result sets with an empty first result set and return second result set, such as stored procedures.	Do not turn off warnings for Freeform SQL statements with empty results, such as updates.
XQuery Success Code	Lets you validate Transaction Services reports that use XQuery.	User-defined.	false

Ignore Empty Result for Freeform SQL

Ignore Empty Result for Freeform SQL is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Ignore Empty Result for Freeform SQL VLDB property provides the flexibility to display or hide warnings when a Freeform SQL statement returns an empty result.

Freeform SQL is intended to be used to return results that can be displayed on a Freeform SQL report. However, Freeform SQL can also be used to execute SQL statements that create tables, update tables, or perform other database maintenance tasks. These types of actions do not return any results and therefore would return a warning when executing a Freeform SQL report. If you routinely use Freeform SQL for these purposes, you can hide these warnings since an empty result set is expected.

This VLDB property has the following options:

- **Do not turn off warnings for Freeform SQL statements with empty results, such as updates** (default): This option allows warnings to be displayed when a Freeform SQL statement causes a Freeform SQL report to return an empty result. This is a good option if you use Freeform SQL to return and display data with Freeform SQL reports.
- **Turn off warnings for Freeform SQL statements with empty results, such as updates**: Select this option to hide all warnings when a Freeform SQL statement causes a Freeform SQL report to return an empty result. This is a good option if you commonly use Freeform SQL to execute various SQL statements that are not expected to return any report results. This prevents users from seeing a warning every time a SQL statement is executed using Freeform SQL.

  However, be aware that if you also use Freeform SQL to return and display data with Freeform SQL reports, no warnings are displayed if the report returns a single empty result.
- **Turn off warnings for Freeform SQL statements that return multiple result sets with an empty first result set and return second result set, such as stored procedures**: Select this option to hide all warnings when a Freeform SQL report returns an initial empty result, followed by additional results that include information. Stored procedures can sometimes have this type of behavior as they can include statements that do not return any results (such as update statements or create table statements), followed by statements to return information from the updated tables. This prevents users from seeing a warning when these types of stored procedures are executed using Freeform SQL.

  If you select this option and a Freeform SQL report returns only a single empty result, then a warning is still displayed.

Levels at which you can set this

Database instance and report

XQuery Success Code

XQuery Success Code is an advanced property that is hidden by default. For instructions on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The XQuery Success Code VLDB property lets you validate Transaction Services reports that use XQuery. MicroStrategy Transaction Services and XQuery allow you to access and update information available in third-party web services data sources. The steps to create

a Transaction Services report using XQuery are provided in the *Advanced Reporting Guide*.

When Transaction Services and XQuery are used to update data for third-party web services, sending the data to be updated is considered as a successful transaction. By default, any errors that occur for the third-party web service during a transaction are not returned to MicroStrategy.

To check for errors, you can include logic in your XQuery syntax to determine if the transaction successfully updated the data within the third-party web service. Just after the XQuery table declaration, you can include the following syntax:

```
<ErrorCode>{Error_Code}</ErrorCode>
<ErrorMessage>{Error_Message}</ErrorMessage>
```

In the syntax above:

- *Error_Code* is a variable that you must define in your XQuery statement to retrieve the success or error code from the third-party web service, for the action that attempts the transaction. The logic to return an error code depends on the third-party web service that you are attempting to perform the transaction on.
- *Error_Message* is either a static error message that you supply, or a variable that you must define in your XQuery statement to retrieve any resulting error message from the third-party web service.

By including this syntax in your XQuery statement, the XQuery Success Code VLDB property is used to validate the transaction. The information returned by the *Error_Code* variable is compared to the value supplied for the XQuery Success Code. By default, the XQuery Success Code is defined as “false”, but you can type any valid string. If the *Error_Code* and XQuery Success Code are identical, then the content in the *Error_Message* is not returned and the transaction is returned as a success. However, if the *Error_Code* returns any value other than the XQuery Success Code, the content for the *Error_Message* is returned. This lets you validate each transaction that is sent to the third-party web service.

Levels at which you can set this

Database instance and report

Limiting report rows, SQL size, and SQL time-out: Governing

The table below summarizes the Governing VLDB properties. Additional details about each property, including examples where necessary, are available by clicking on the links in the table.

Property	Description	Possible Values	Default Value
Autocommit	Determines whether a commit statement is	• ON	ON

Property	Description	Possible Values	Default Value
	automatically issued after each SQL statement for a database connection.	• OFF	
Intermediate Row Limit	The maximum number of rows returned to the server for each intermediate pass. (0 = unlimited number of rows; -1 = use value from higher level.)	User-defined	-1 (Use value from higher level)
Maximum SQL/MDX Size	Maximum size of SQL string accepted by ODBC driver (bytes).	User-defined	65536
Results Set Row Limit	The maximum number of rows returned to the Server for the final result set. (0 = unlimited number of rows; -1 = use value from higher level.)	User-defined	-1 (Use value from higher level)
Limiting report rows, SQL size, and SQL time-out: Governing	Single SQL pass time-out in seconds. (0 = time limit not enforced by this governing setting)	User-defined	0 (Time limit not enforced by this governing setting)

Autocommit

The Autocommit VLDB property determines whether a commit statement is automatically issued after each SQL statement for a database connection. You have the following options:

- **ON: A commit is automatically issued after each SQL statement by the database connection**: By default, a commit is issued automatically after each SQL statement. This allows you to query a database without having to manually issue commit statements and other required transaction control commands.
- **OFF: No commit is automatically issued after each SQL statement by the database connection**: Commit statements are not issued automatically after each SQL statement.

  Multiple SQL statements are required for various reporting and analysis features in MicroStrategy. When multiple SQL statements are used, each can be viewed as a separate transaction. If your database is being updated by a separate transaction, ETL process, or other update, this can cause data inconsistency with each SQL statement, since each SQL statement is returned as a separate transaction. Disabling automatic commit statements includes all SQL statements as a single transaction, which can be used in conjunction with other database techniques to ensure data consistency when reporting and analyzing a database that is being updated. For example, if reporting on an Oracle database you can use this in conjunction with defining the isolation level of the SQL statements.

  Be aware that if you disable automatic commit statements for each SQL statement, these transaction control commands must be included for the report. If you are using Freeform SQL or creating your own SQL statement for use in MicroStrategy,

these can be included directly in those SQL statements. For reports that use SQL that is automatically generated by MicroStrategy, you can use the Pre/Post Statement VLDB properties (see *Customizing SQL statements: Pre/Post Statements, page 117*) to provide the required transaction control commands.

Levels at which you can set this

Project and report

Intermediate Row Limit

The Intermediate Row Limit property is used to limit the number of rows of data returned to the Server from pure SELECT statements issued apart from the final pass. Apart from the final pass, pure SELECT statements are usually executed if there are analytical functions or partition pre-queries to process. Since the partition pre-queries return only a handful of rows, the SELECT statements issued for analytical function processing decide the number of rows set in most cases. If the limit is exceeded, the report fails with an error message. When it is set to the default, the Intermediate Row Limit takes the value of the Result Set Row Limit VLDB property at the report (highest) level.

The table below explains the possible values and their behavior:

Value	Behavior
0	No limit on number of rows returned
Number	Number of rows returned is limited to the specified number

Levels at which you can set this

Report only

Maximum SQL/MDX Size

The Maximum SQL/MDX Size property specifies the SQL size (in bytes) on a pass-by-pass basis. If the limit is exceeded, the report execution is terminated and an error message is returned. The error message usually mentions that a SQL/MDX string is longer than a corresponding limitation. The limit you choose should be based on the size of the SQL string accepted by your ODBC driver.

The table below explains the possible values and their behavior:

Value	Behavior
0	No limit on SQL pass size

Value	Behavior
Number	The maximum SQL pass size (in bytes) is limited to the specified number
Default	By selecting the check box **Use default inherited value**, the value is set to the default for the database type used for the related database instance. The default size varies depending on the database type.

Increasing the maximum to a large value can cause the report to fail in the ODBC driver. This is dependent on the database type you are using.

Levels at which you can set this

Database instance only

Results Set Row Limit

The Results Set Row Limit property is used to limit the number of rows returned from the final results set SELECT statements issued. This property is report-specific.

- If the report result set exceeds the limit specified in the **Result Set Row Limit**, the report execution is terminated.
- This property overrides the Number of report result rows setting in the Project Configuration Editor: Governing Rules category. For details on this project setting, see the *MicroStrategy Developer Help* (formerly the *MicroStrategy Desktop Help*).

When the report contains a custom group, this property is applied to each element in the group. Therefore, the final result set displayed could be larger than the predefined setting. For example, if you set the Result Set Row Limit to 1,000, it means you want only 1,000 rows to be returned. Now apply this setting to each element in the custom group. If the group has three elements and each uses the maximum specified in the setting (1,000), the final report returns 3,000 rows.

The table below explains the possible values and their behavior:

Value	Behavior
0	No limit on the number of result rows
Number	Number of rows returned limited to the specified number

Levels at which you can set this

Report only

SQL Time Out (Per Pass)

The SQL Time Out property is used to avoid lengthy intermediate passes. If any pass of SQL runs longer than the set time (in seconds), the report execution is terminated.

The table below explains the possible values and their behavior:

Value	Behavior
0	This governing setting does not impose a time limit on SQL pass execution.
Number	The maximum amount of time (in seconds) a SQL pass can execute is limited to the specified number.

Levels at which you can set this

Database instance and report

Retrieving data: Indexing

The table below summarizes the Indexing VLDB properties. Additional details about each property, including examples where necessary, are available by clicking on the links in the table.

Property	Description	Possible Values	Default Value
Allow Index on Metric	Determines whether or not to allow the creation of indexes on fact or metric columns.	• Don't allow the creation of indexes on metric columns • Allow the creation of indexes on metric columns (if the Intermediate Table Index setting is set to create)	Don't allow the creation of indexes on metric columns
Index Post String and Index Qualifier	Defines the string that is appended at the end of the CREATE INDEX statement. For example: IN INDEXSPACE	User-defined	NULL
Retrieving data: Indexing	Defines the prefix to use when automatically creating indexes for intermediate SQL passes. The prefix is added to the beginning of the CREATE INDEX statement.	User-defined	NULL

Property	Description	Possible Values	Default Value
Index Post String and Index Qualifier	Defines the string to parse in between the CREATE and INDEX words. For example: CLUSTERED	User-defined	NULL
Intermediate Table Index	Determines whether and when to create an index for the intermediate table.	• Don't create an index • Create partitioning key (typically applicable to MPP systems) • Create partitioning key and secondary index on intermediate table • Create only secondary index on intermediate table	Don't create an index
Max Columns in Column Placeholder	Determines the maximum number of columns that replace the column wildcard ("!!!") in pre and post statements. 0 = all columns (no limit).	User-defined	0 (No limit)
Max Columns in Index	Determines the maximum number of columns that can be included in partition key or index.	User-defined	No limit
Primary Index Type	Determines whether a primary key is created instead of a partitioning key for databases that support both types, such as UDB.	• Create primary key (where applicable) if the intermediate table index setting is set to create a primary index. • Create primary index/partitioning key (where applicable) if the intermediate table index setting is set to create a primary index.	Create primary key (where applicable) if the intermediate table index setting is set to create a primary index.
Secondary Index Order	Defines whether an index is created before or after inserting data into a table.	• Create index after inserting into table • Create index before inserting into table	Create index after inserting into table
Secondary Index Type	Defines what type of index is created for temporary table column indexing.	• Create Composite Index for Temporary Table Column Indexing • Create Individual Indexes for Temporary Table Column Indexing	Create Composite Index for Temporary Table Column Indexing

Allow Index on Metric

Allow Index on Metric is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Allow Index on Metric property determines whether or not to use fact or metric columns in index creation. You can see better performance in different environments, especially in Teradata, when you add the fact or metric column in the index. Usually, the indexes are created on attribute columns; but with this setting, the fact or metric columns are added as well. All fact or metric columns are added.

Levels at which you can set this

Database instance, report, and template

Example

Do not allow creation of indexes on fact or metric columns (default)

```
create table ZZT8L005Y1YEA000 (
                CATEGORY_ID BYTE,
                REGION_ID BYTE,
                YEAR_ID SHORT,
                WJXBFS1 DOUBLE,
                WJXBFS2 DOUBLE)
insert into ZZT8L005Y1YEA000
select a13.[CATEGORY_ID] AS CATEGORY_ID,
                a15.[REGION_ID] AS REGION_ID,
                a16.[YEAR_ID] AS YEAR_ID,
                sum((a11.[QTY_SOLD] * (a11.[UNIT_PRICE] -
a11.[DISCOUNT]))) as WJXBFS1,
                sum((a11.[QTY_SOLD] * ((a11.[UNIT_PRICE] -
a11.[DISCOUNT]) - a11.[UNIT_COST]))) as
WJXBFS2
from [ORDER_DETAIL] a11,
                [LU_ITEM] a12,
                [LU_SUBCATEG] a13,
                [LU_EMPLOYEE] a14,
                [LU_CALL_CTR] a15,
                [LU_DAY] a16
where a11.[ITEM_ID] = a12.[ITEM_ID] and
                a12.[SUBCAT_ID] = a13.[SUBCAT_ID] and
                a11.[EMP_ID] = a14.[EMP_ID] and
                a14.[CALL_CTR_ID] = a15.[CALL_CTR_ID] and
                a11.[ORDER_DATE] = a16.[DAY_DATE]
 and a15.[REGION_ID] in (1)
group by a13.[CATEGORY_ID],
                a15.[REGION_ID],
                a16.[YEAR_ID]
create index ZZT8L005Y1YEA000_i on ZZT8L005Y1YEA000
(CATEGORY_ID, REGION_ID, YEAR_ID)
```

Allow the creation of indexes on fact or metric columns

This example is the same as the example above except that the last line of code should be replaced with the following:

```
create index ZZT8L005YAGEA000_i on ZZT8L005YAGEA000
(CATEGORY_ID, REGION_ID, YEAR_ID, WJXBFS1, WJXBFS2)
```

Index Prefix

This property allows you to define the prefix to add to the beginning of the CREATE INDEX statement when automatically creating indexes for intermediate SQL passes.

For example, the index prefix you define appears in the CREATE INDEX statement as shown below:

```
create index(index prefix)
IDX_TEMP1(STORE_ID, STORE_DESC)
```

Levels at which you can set this

Database instance, report, and template

Index Post String and Index Qualifier

The Index Post String and Index Qualifier property can be used to customize the CREATE INDEX statement. Indexes can be created when the Intermediate Table Type is set to Permanent Tables, Temporary Tables, and Views (most platforms do not support indexes on views). These two settings can be used to specify the type of index to be created and the storage parameters as provided by the specific database platform. If the Index Post String and Index Qualifier are set to a certain string, then for all the CREATE INDEX statements, the Index Post String and Index Qualifier are applied.

The create index syntax pattern is as follows:

- All platforms except Teradata:

  ```
  create <<Index Qualifier>> index i_[Table Name] on
  [Table Name] ([Column List]) <<Index Post String>>
  ```

- Teradata:

  ```
  create <<Index Qualifier>> index i_[Table Name]
  ([Column List]) on [Table Name] <<Index Post String>>
  ```

Example

Index Post String

The Index Post String setting allows you to add a custom string to the end of the CREATE INDEX statement.

```
Index Post String = /* in tablespace1 */
create index IDX_TEMP1(STORE_ID, STORE_DESC) /* in
"tablespace1*/
```

Levels at which you can set this

Database instance, report, and template

Intermediate Table Index

The Intermediate Table Index property is used to control the primary and secondary indexes generated for platforms that support them. This property is for permanent tables and temporary tables, where applicable. In the VLDB Properties Editor, select an option to view example SQL statements used by various databases for the selected option:

- **Don't create an index** (default)
- **Create partitioning key (typically applicable to MPP systems)**
- **Create portioning key and secondary index on intermediate table**
- **Create only secondary index on intermediate table**

Levels at which you can set this

Database instance, report, and template

Max Columns in Column Placeholder

Max Columns in Column Placeholder is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Max Columns in Column Placeholder property controls the maximum number of columns that replace the column wildcard ("!!!") in pre and post statements. This limit applies to both the primary and the secondary indexes.

The table below explains the possible values and their behavior:

Value	Behavior
0	All attribute ID columns go into the index
Number	The maximum number of attribute ID columns to use with the wildcard

Levels at which you can set this

Database instance only

Max Columns in Index

The Max Columns in Index property controls the maximum number of columns that can be used when creating an index. This limit applies to both primary and secondary indexes. If the maximum is five columns but there are 10 columns available to index, the first five are selected. However, each attribute has a "weight" that you can set. When SQL is generated, the attributes are selected in ascending order of "weight." By combining Attribute Weights and the Max Columns in Index properties, you can designate any attribute to be included in the index.

You can define attribute weights in the Project Configuration Editor. Select the **Report definition: SQL generation** category, and in the Attribute weights section, click **Modify**.

The table below explains the possible values and their behavior:

Value	Behavior
0	All attribute ID columns are placed in the index
Number	The maximum number of attribute ID columns that are placed in the index

Levels at which you can set this

Database instance, report, and template

Primary Index Type

Primary Index Type is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Primary Index Type property determines the pattern for creating primary keys and indexes. In the VLDB Properties Editor, select an option to view example SQL statements used by various databases for the selected option. The examples also display whether the option is applicable for a given database type. If you select an option that is not applicable for the database type that you use, then the other option is used automatically. While this ensures that the primary index type is correct for your database, you should select an option that is listed as applicable for the database that you use.

Some databases such as DB2 UDB support both primary index type options. Use the example SQL statements and your third-party database documentation to determine the best option for your environment.

Levels at which you can set this

Database instance, report, and template

Secondary Index Order

The Secondary Index Order VLDB property allows you to define whether an index is created before or after inserting data into a table. This VLDB property has the following options:

- **Create index after inserting into table** (default): This option creates the index after inserting data into a table, which is a good option to support most database and indexing strategies.

- **Create index before inserting into table**: This option creates the index before inserting data into a table, which can improve performance for some environments, including Sybase IQ. The type of index created can also help to improve performance in these types of environments, and can be configured with the Secondary Index Type VLDB property (see *Secondary Index Order, page 58*).

Levels at which you can set this

Database instance, report, and template

Secondary Index Type

The Secondary Index Type VLDB property allows you to define what type of index is created for temporary table column indexing. This VLDB property has the following options:

- **Create Composite Index for Temporary Table Column Indexing** (default): This option creates composite indexes for temporary table column indexing. This is a good option to support most database and indexing strategies.
- **Create Individual Indexes for Temporary Table Column Indexing**: This option creates individual indexes for temporary table column indexing. This can improve performance for some environments, including Sybase IQ. The order in which the index is created can also help to improve performance in these types of environments, and can be configured with the Secondary Index Order VLDB property (see *Secondary Index Type, page 59*).

Levels at which you can set this

Database instance, report, and template

Relating column data with SQL: Joins

The table below summarizes the Joins VLDB properties. Additional details about each property, including examples where necessary, are available by clicking on the links in the table.

Property	Description	Possible Values	Default Value
Attribute to join when key from neither side can be supported by the other side	Controls whether tables are joined only on the common keys or on all common columns for each table.	• Join common key on both sides • Join common attributes (reduced) on both sides	Join common key on both sides

Property	Description	Possible Values	Default Value
Base Table Join for Template	Controls whether two fact tables are directly joined together. If you choose Temp Table Join, the Analytical Engine calculates results independently from each fact table and places those results into two intermediate tables. These intermediate tables are then joined together.	• Temp table join • Fact table join	Temp table join
Cartesian Join Evaluation	Allows the MicroStrategy SQL Engine to use a new algorithm for evaluating whether or not a Cartesian join is necessary.	• Do not reevaluate cartesian joins • Reevaluate cartesian joins	Do not reevaluate cartesian joins
Cartesian Join Warning	Action that occurs when the Analytical Engine generates a report that contains a Cartesian join.	• Execute • Cancel execution • Cancel execution only when warehouse table is involved in either side of cartesian join • If only one side of cartesian join contains warehouse tables, SQL will be executed without warning	Execute
Downward Outer Join Option	Allows users to choose how to handle metrics which have a higher level than the template.	• Do not preserve all the rows for metrics higher than template level • Preserve all the rows for metrics higher than template level w/o report filter • Preserve all the rows for metrics higher than template level with report filter • Do not do downward outer join for database that support full outer join • Do not do downward outer join for database that support full outer join, and order temp tables in last pass by level	Do not preserve all the rows for metrics higher than template level

Property	Description	Possible Values	Default Value
DSS Star Join	Controls which lookup tables are included in the join against the fact table. For a partial star join, the Analytical Engine joins the lookup tables of all attributes present in either the template or the filter or metric level, if needed.	• No star join • Partial star join	No star join
From Clause Order	Determines whether to use the normal FROM clause order as generated by the Analytical Engine or to switch the order.	• Normal FROM clause order as generated by the engine • Move last table in normal FROM clause order to the first • Move MQ table in normal From clause order to the last (for RedBrick) • Reverse FROM clause order as generated by the engine	Normal FROM clause order as generated by the engine
Full Outer Join Support	Indicates whether the database platform supports full outer joins.	• No support • Support	No support
Join Type	Type of column join.	• Join 89 • Join 92 • SQL 89 Inner Join and Cross Join and SQL 92 Outer Join • SQL 89 Inner Join and SQL 92 Outer Join and Cross Join	Join 89
Lookup Table Join Order	Determines how lookup tables are loaded for join operations.	• Partially based on attribute level (behavior prior to version 8.0.1) • Fully based on attribute level. Lookup tables for lower level attributes are joined before those for higher level attributes	Partially based on attribute level (behavior prior to version 8.0.1)
Max Tables in Join	Maximum number of tables to join together.	• User-defined	No limit

Property	Description	Possible Values	Default Value
Max Tables in Join Warning	Action that occurs when the Analytical Engine generates a report that exceeds the maximum number of tables in the join limit.	• Execute • Cancel execution	Cancel execution
Nested Aggregation Outer Joins	Defines when outer joins are performed on metrics that are defined with nested aggregation functions.	• Do not perform outer join on nested aggregation • Do perform outer join on nested aggregation when all formulas have the same level • Do perform downward outer join on nested aggregation when all formulas can downward outer join to a common lower level	Do not perform outer join on nested aggregation
Preserving data using outer joins			
Preserve all final pass result elements	Perform an outer join to the final result set in the final pass.	• Preserve common elements of final pass result table and lookup/relationship table • Preserve all final result pass elements • Preserve all elements of final pass result table with respect to lookup table but not relationship table • Do not listen to per report level setting, preserve elements of final pass according to the setting at attribute level. If this choice is selected at attribute level, it will be treated as preserve common elements (i.e. choice 1)	Preserve common elements of final pass result table and lookup/relationship table.

Property	Description	Possible Values	Default Value
Preserve all lookup table elements	Perform an outer join to lookup table in the final pass.	• Preserve common elements of lookup and final pass result table • Preserve lookup table elements joined to final pass result table based on fact table keys • Preserve lookup table elements joined to final pass result table based on template attributes without filter • Preserve lookup table elements joined to final pass result table based on template attributes with filter	Preserve common elements of lookup and final pass result table

Attribute to join when key from neither side can be supported by the other side

The Attribute to join when key from neither side can be supported by the other side is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property determines how MicroStrategy joins tables with common columns. The options for this property are:

- **Join common key on both sides** (default): Joins on tables only use columns that are in each table, and are also keys for each table.
- **Join common attributes (reduced) on both sides**: Joins between tables use all common attribute columns to perform the join. This functionality can be helpful in a couple of different scenarios.
    - You have two different tables named Table1 and Table2. Both tables share 3 ID columns for Year, Month, and Date along with other columns of data. Table1 uses Year, Month, and Date as keys while Table2 uses only Year and Month as keys. Since the ID column for Date is not a key for Table2, you must set this option to include Day to join the tables along with Year and Month.
    - You have a table named Table1 that includes the columns for the attributes Quarter, Month of Year, and Month. Since Month is a child of Quarter and Month of Year, its ID column is used as the key for Table1. There is also a temporary table named TempTable that includes the columns for the attributes Quarter, Month of Year, and Year, using all three ID columns as keys of the table. It is not possible to join Table1 and TempTable unless you set this option because they do not share any common keys. If you set this option, Table1 and TempTable can join on the common attributes Quarter and Month of Year.

Levels at which you can set this

Database instance, report, and template

Base Table Join for Template

The Base Table Join for Template is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

When reports contain metrics from different fact tables or a compound metric made up of data from different fact tables, then the Base Table Join for Template property can be used to choose between intermediate table joins and base tables joins. The property is mainly performance-related. If intermediate table join is chosen, then the type of intermediate table is governed by the Intermediate Table Type VLDB property (see *Intermediate Table Type, page 219* in the Table Properties section).

Caution must be taken when changing this setting since the results can be different depending on the types of metrics on the report.

Levels at which you can set this

Database instance, report, and template

Example

Use Temp Table Join (default)

```
select a11.MARKET_NBR MARKET_NBR,
sum(a11.CLE_SLS_DLR)
                CLEARANCESAL
into #ZZTIS00H5D3SP000
from HARI_MARKET_DIVISION a11
group by a11.MARKET_NBR
select a11.MARKET_NBR MARKET_NBR,
sum(a11.COST_AMT)
                COSTAMOUNT
into #ZZTIS00H5D3SP001
from HARI_COST_MARKET_DIV a11
group by a11.MARKET_NBR
select pa1.MARKET_NBR MARKET_NBR,
                a11.MARKET_DESC MARKET_DESC,
                pa1.CLEARANCESAL WJXBFS1,
                pa2.COSTAMOUNT WJXBFS2
from #ZZTIS00H5D3SP000 pa1
        left outer join #ZZTIS00H5D3SP001 pa2
          on (pa1.MARKET_NBR = pa2.MARKET_NBR)
        left outer join HARI_LOOKUP_MARKET a11
          on (pa1.MARKET_NBR = a11.MARKET_NBR)
```

Use Fact Table Join

```
select a11.MARKET_NBR MARKET_NBR,
                max(a13.MARKET_DESC) MARKET_DESC,
                sum(a12.CLE_SLS_DLR) CLEARANCESAL,
                sum(a11.COST_AMT) COSTAMOUNT
from HARI_COST_MARKET_DIV a11
```

```
        join HARI_MARKET_DIVISION a12
          on (a11.CUR_TRN_DT = a12.CUR_TRN_DT
        and a11.DIVISION_NBR = a12.DIVISION_NBR
        and a11.MARKET_NBR = a12.MARKET_NBR)
        join HARI_LOOKUP_MARKET a13
          on (a11.MARKET_NBR = a13.MARKET_NBR)
group by a11.MARKET_NBR
```

Cartesian Join Evaluation

Cartesian Join Evaluation is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This property allows the MicroStrategy SQL Engine to use a new algorithm for evaluating whether or not a Cartesian join is necessary. The new algorithm can sometimes avoid a Cartesian join when the old algorithm cannot. For backward compatibility, the default is the old algorithm. If you see Cartesian joins that appear to be avoidable, use this property to determine whether the engine's new algorithm avoids the Cartesian join.

Levels at which you can set this

Database instance, report, and template

Examples

Do Not Reevaluate Cartesian Joins (default)

```
select a12.ATTR1_ID ATTR1_ID,
max(a12.ATTR1_DESC) ATTR1_DESC,
a13.ATTR2_ID ATTR2_ID,
max(a13.ATTR2_DESC) ATTR2_DESC,
count(a11.FACT_ID) METRIC
from FACTTABLE a11
cross join LU_TABLE1 a12
join LU_TABLE2 a13
on (a11.ATTR3_ID = a13.ATTR3_ID and
a12.ATTR1_ID = a13.ATTR1_CD)
group by a12.ATTR1_ID,
a13.ATTR2_ID
```

Reevaluate the Cartesian Joins

```
select a12.ATTR1_ID ATTR1_ID,
max(a12.ATTR1_DESC) ATTR1_DESC,
a13.ATTR2_ID ATTR2_ID,
max(a13.ATTR2_DESC) ATTR2_DESC,
count(a11.FACT_ID) METRIC
from FACTTABLE a11
join LU_TABLE2 a13
on (a11.ATTR3_ID = a13.ATTR3_ID)
join LU_TABLE1 a12
on (a12.ATTR1_ID = a13.ATTR1_CD)
group by a12.ATTR1_ID,
a13.ATTR2_ID
```

Cartesian Join Warning

Cartesian joins are usually costly to perform. However, a Cartesian join of two warehouse tables is much more costly than a Cartesian join of two intermediate tables.

- **Execute** (default): When any Cartesian join is encountered, execution continues without warning.
- **Cancel execution**: When a report contains any Cartesian join, execution is canceled.
- **Cancel execution only when warehouse table is involved in either side of Cartesian join**: The execution is canceled only when a warehouse table is involved in a Cartesian join. In other words, the Cartesian join is allowed when all tables involved in the join are intermediate tables.
- **If only one side of Cartesian join contains warehouse tables, SQL will be executed without warning**: When all tables involved in the Cartesian join are intermediate tables, the SQL is executed without warning. This option also allow a Cartesian join if a warehouse table is only on one side of the join and cancels it if both sides are warehouse tables.

- In the rare situation when a warehouse table is Cartesian-joined to an intermediate table, the execution is usually canceled. However, there may be times when you want to allow this to execute. In this case, you can choose the option: **If only one side of Cartesian join contains warehouse tables, SQL will be executed without warning.** If this option is selected, the execution is canceled only when warehouse tables are involved in both sides of the Cartesian join.
- Some Cartesian joins may not be a direct table-to-table join. If one join "Cartesian joins" to another join, and one of the joins contains a warehouse table (not an intermediate table), then the execution is either canceled or allowed depending on the option selected (see below). For example, if (TT_A join TT_B) Cartesian join (TT_C join WH_D) the following occurs based on the following settings:
- If the setting **Cancel execution only when warehouse table is involved in Cartesian join** is selected, execution is canceled. In the above example, execution is canceled because a warehouse table is used, even though TT_A, TT_B, and TT_C are all intermediate tables.
- If the setting **If only one side of Cartesian…** is selected, SQL runs without warning. In the above example, execution continues because a warehouse table (WH_D) is used on only one side of the join.

Levels at which you can set this

Database instance, report, and template

Downward Outer Join Option

Downward Outer Join Option is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

To understand Downward Outer Join, consider the following report that contains the attribute Store and two metrics, Sales Per Store (M1) and Inventory Per Region (M2). The attribute Region is a parent of Store. Both M1 and M2 are set to Outer Join.

Store	Sales Per Store (M1)	Inventory Per Region (M2)

Traditionally, the outer join flag is ignored, because M2 (at Region level) is higher than the report level of Store. It is difficult to preserve all of the stores for a metric at the Region level. However, you can preserve rows for a metric at a higher level than the report. Since M2 is at the region level, it is impossible to preserve all regions for M2 because the report only shows Store. To do that, a downward join pass is needed to find all stores that belong to the region in M2, so that a union is formed among all these stores with the stores in M1.

When performing a downward join, another issue arises. Even though all the stores that belong to the region in M2 can be found, these stores may not be those from which M2 is calculated. If a report filters on a subset of stores, then M2 (if it is a filtered metric) is calculated only from those stores, and aggregated to regions. When a downward join is done, either all the stores that belong to the regions in M2 are included or only those stores that belong to the regions in M2 and in the report filter. Hence, this property has three options.

Levels at which you can set this

Database instance, report, and template

Example

Using the above example and applying a filter for Atlanta and Charlotte, the default **Do not preserve all the rows for metrics higher than template level** option returns the following results. Note that Charlotte does not appear because it has no sales data in the fact table; the outer join is ignored. The outer join flag on metrics higher than template level is ignored.

Store	Sales Per Store (M1)	Inventory Per Region (M2)
Atlanta	100	300

Using **Preserve all the rows for metrics higher than template level without report filter** returns the results shown below. Now Charlotte appears because the outer join is used, and it has an inventory, but Washington appears as well because it is in the Region, and the filter is not applied.

Store	Sales Per Store (M1)	Inventory Per Region (M2)
Atlanta	100	300
Charlotte		300
Washington		300

Using **Preserve all the rows for metrics higher than template level with report filter** produces the following results. Washington is filtered out but Charlotte still appears because of the outer join.

Store	Sales Per Store (M1)	Inventory Per Region (M2)
Atlanta	100	300
Charlotte		300

For backward compatibility, the default is to ignore the outer join flag for metrics higher than template level. This is the SQL Engine behavior for MicroStrategy 6.x or lower, as well as for MicroStrategy 7.0 and 7.1.

DSS Star Join

DSS Star Join is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The DSS Star Join property specifies whether a partial star join is performed or not. A partial star join means the lookup table of a column is joined if and only if a column is in the SELECT clause or involved in a qualification in the WHERE clause of the SQL. In certain databases, for example, RedBrick and Teradata, partial star joins can improve SQL performance if certain types of indexes are maintained in the data warehouse. Notice that the lookup table joined in a partial star join is not necessarily the same as the lookup table defined in the attribute form editor. Any table that acts as a lookup table rather than a fact table in the SQL and contains the column is considered a feasible lookup table.

Levels at which you can set this

Database instance, report, and template

Examples

No Star Join (default)

```
select distinct a11.PBTNAME PBTNAME
from STORE_ITEM_PTMAP a11
where a11.YEAR_ID in (1994)
select a11.ITEM_NBR ITEM_NBR,
        a11.CLASS_NBR  CLASS_NBR,
        a13.ITEM_DESC  ITEM_DESC,
        a13.CLASS_DESC  CLASS_DESC,
        a11.STORE_NBR  STORE_NBR,
        a14.STORE_DESC  STORE_DESC,
        sum(a11.REG_SLS_DLR)  WJXBFS1
from STORE_ITEM_94 a11,
        LOOKUP_DAY a12,
        LOOKUP_ITEM a13,
        LOOKUP_STORE a14
where a11.CUR_TRN_DT = a12.CUR_TRN_DT and
        a11.CLASS_NBR = a13.CLASS_NBR and
        a11.ITEM_NBR = a13.ITEM_NBR and
        a11.STORE_NBR = a14.STORE_NBR
 and    a12.YEAR_ID in (1994)
group by a11.ITEM_NBR,
        a11.CLASS_NBR,
        a13.ITEM_DESC,
        a13.CLASS_DESC,
        a11.STORE_NBR,
        a14.STORE_DESC
```

Partial Star Join

```
select distinct a11.PBTNAME  PBTNAME
from STORE_ITEM_PTMAP a11,
        LOOKUP_YEAR a12
where a11.YEAR_ID = a12.YEAR_ID
 and    a11.YEAR_ID in (1994)
Pass1 - Duration: 0:00:00.49
select a11.ITEM_NBR  ITEM_NBR,
        a11.CLASS_NBR  CLASS_NBR,
        a13.ITEM_DESC  ITEM_DESC,
        a13.CLASS_DESC  CLASS_DESC,
        a11.STORE_NBR  STORE_NBR,
        a14.STORE_DESC  STORE_DESC,
        sum(a11.REG_SLS_DLR)  WJXBFS1
from STORE_ITEM_94 a11,
        LOOKUP_DAY a12,
        LOOKUP_ITEM a13,
        LOOKUP_STORE a14
where a11.CUR_TRN_DT = a12.CUR_TRN_DT and
        a11.CLASS_NBR = a13.CLASS_NBR and
        a11.ITEM_NBR = a13.ITEM_NBR and
        a11.STORE_NBR = a14.STORE_NBR
and      a12.YEAR_ID in (1994)
group by a11.ITEM_NBR,
        a11.CLASS_NBR,
        a13.ITEM_DESC,
        a13.CLASS_DESC,
        a11.STORE_NBR,
        a14.STORE_DESC
```

From Clause Order

Some database platforms, such as Oracle and RedBrick, perform better depending on the order of the tables in the FROM clause. The FROM Clause Ordering property alters the order that the tables appear in the FROM clause. The MicroStrategy SQL Engine

normally puts the fact table first in the FROM clause. When the property is set to switch the FROM clause order, the fact table is moved to the second table in the clause. However, if there are two fact tables in the FROM clause, it simply switches the order of the two tables.

Levels at which you can set this

Database instance, report, and template

Examples

Normal FROM clause order as generated by the engine

```
select a12.CUSTOMER_ID CUSTOMER_ID,
                sum(a11.ORDER_AMT) WJXBFS1
from ORDER_FACT a11
                join LU_ORDER a12
                  on (a11.ORDER_ID = a12.ORDER_ID)
group by a12.CUSTOMER_ID
```

Switch FROM clause order as generated by the engine

```
select a12.CUSTOMER_ID CUSTOMER_ID,
                sum(a11.ORDER_AMT) WJXBFS1
from LU_ORDER a12
                join ORDER_FACT a11
                  on (a11.ORDER_ID = a12.ORDER_ID)
group by a12.CUSTOMER_ID
```

Move MQ Table in normal FROM clause order to the last (for RedBrick)

This setting is added primarily for RedBrick users. The default order of table joins is as follows:

1. Join the fact tables together.
2. Join the metric qualification table.
3. Join the relationship table.
4. Join the lookup tables if needed.

This option changes the order to the following:

1. Join the fact tables together.
2. Join the relationship table.
3. Join the lookup tables.
4. Join the metric qualification table.

Full Outer Join Support

Full Outer Join Support is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Full Outer Join Support property specifies whether the database platform supports full outer join syntax:

- **No support** (default): Full outer joins are not supported or processed to return results. This can help to prevent costly outer join queries and also avoids errors for databases that do not support full outer joins. Additionally, if your database does not support the `COALESCE` function, you should set this property to No support.
- **Support**: Full outer joins are attempted when required by your reporting and dashboarding actions. By selecting this option, the Join Type VLDB property is assumed to be Join 92 and any other setting in Join Type is ignored. Additionally, the `COALESCE` function can be included in the SQL query.

  Since full outer joins can require a lot of database and Intelligence Server resources, and full outer joins are not supported for all databases, it is recommended to enable support for individual reports first. If your results are returned successfully and full outer joins are used often for your reporting and dashboarding environment, you can consider enabling support for the entire database. However, enabling full outer join support for specific reports is recommended if full outer joins are only used for a small to moderate amount of reporting needs. Creating a template with full outer join support enabled can save report developers time when requiring full outer joins.

Levels at which you can set this

Database instance, report, and template

Examples

Full Outer Join Not Supported (default)

```
select a12.YEAR_ID YEAR_ID,
                sum(a11.TOT_SLS_DLR) TOTALSALESCO
into #ZZTIS00H5MJMD000
from HARI_REGION_DIVISION a11
                join HARI_LOOKUP_DAY a12
                  on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
where a12.MONTH_ID = 199411
group by a12.YEAR_ID
select a12.YEAR_ID YEAR_ID,
                sum(a11.TOT_SLS_DLR) TOTALSALESCO
into #ZZTIS00H5MJMD001
from HARI_REGION_DIVISION a11
                join HARI_LOOKUP_DAY a12
                  on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
where a12.MONTH_ID = 199311
group by a12.YEAR_ID
select pa1.YEAR_ID YEAR_ID
into #ZZTIS00H5MJOJ002
from #ZZTIS00H5MJMD000 pa1
union
select pa2.YEAR_ID YEAR_ID
from #ZZTIS00H5MJMD001 pa2
```

```
select distinct pa3.YEAR_ID YEAR_ID,
                a11.YEAR_DESC YEAR_DESC,
                pa1.TOTALSALESCO TOTALSALESCO,
                pa2.TOTALSALESCO TOTALSALESCO1
from #ZZTIS00H5MJOJ002 pa3
                left outer join #ZZTIS00H5MJMD000 pa1
                  on (pa3.YEAR_ID = pa1.YEAR_ID)
                left outer join #ZZTIS00H5MJMD001 pa2
                  on (pa3.YEAR_ID = pa2.YEAR_ID)
                left outer join HARI_LOOKUP_YEAR a11
                  on (pa3.YEAR_ID = a11.YEAR_ID)
```

Full Outer Join Supported

```
select a12.YEAR_ID YEAR_ID,
                sum(a11.TOT_SLS_DLR) TOTALSALESCO
into #ZZTIS00H5MKMD000
from HARI_REGION_DIVISION a11
                join HARI_LOOKUP_DAY a12
                  on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
where a12.MONTH_ID = 199411
group by a12.YEAR_ID
select a12.YEAR_ID YEAR_ID,
                sum(a11.TOT_SLS_DLR) TOTALSALESCO
into #ZZTIS00H5MKMD001
from HARI_REGION_DIVISION a11
                join HARI_LOOKUP_DAY a12
                  on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
where a12.MONTH_ID = 199311
group by a12.YEAR_ID
select distinct coalesce(pa1.YEAR_ID,
pa2.YEAR_ID) YEAR_ID,
                a11.YEAR_DESC YEAR_DESC,
                pa1.TOTALSALESCO TOTALSALESCO,
                pa2.TOTALSALESCO TOTALSALESCO1
from #ZZTIS00H5MKMD000 pa1
                full outer join #ZZTIS00H5MKMD001 pa2
                  on (pa1.YEAR_ID = pa2.YEAR_ID)
                left outer join HARI_LOOKUP_YEAR a11
on (coalesce(pa1.YEAR_ID, pa2.YEAR_ID) = a11.YEAR_ID)
```

Join Type

The Join Type property determines which ANSI join syntax pattern to use. Some databases, such as Oracle, do not support the ANSI 92 standard yet. Some databases, such as DB2, support both Join 89 and Join 92. Other databases, such as some versions of Teradata, have a mix of the join standards and therefore need their own setting.

MicroStrategy uses different defaults for the join type based on the database you are using. This is to support the most common scenarios for your databases. When selecting a different join type than the default, it is recommended to test this with a report rather than the entire database. By using this strategy you can determine if the join type functions correctly for your database while also providing the required performance.

If the Full Outer Join Support VLDB property (see *Join Type, page 72*) is set to Support, this property is ignored and the Join 92 standard is used.

Levels at which you can set this

Database instance, report, and template

Examples

Join 89 (default)

```
select a22.STORE_NBR STORE_NBR,
max(a22.STORE_DESC) STORE_DESC,
a21.CUR_TRN_DT CUR_TRN_DT,
sum(a21.REG_SLS_DLR) WJXBFS1
from STORE_DIVISION a21,
LOOKUP_STORE a22
where a21.STORE_NBR = a22.STORE_NBR
group by a22.STORE_NBR,
a21.CUR_TRN_DT
```

Join 92

```
select a21.CUR_TRN_DT CUR_TRN_DT,
a22.STORE_NBR STORE_NBR,
max(a22.STORE_DESC) STORE_DESC,
sum(a21.REG_SLS_DLR) WJXBFS1
from STORE_DIVISION a21
join LOOKUP_STORE a22
on (a21.STORE_NBR = a22.STORE_NBR)
group by a21.CUR_TRN_DT,
a22.STORE_NBR
```

SQL 89 Inner Join and SQL 92 Outer Join

```
create table ZZOL00 as
select a23.STORE_NBR STORE_NBR,
a23.MARKET_NBR MARKET_NBR,
a22.DEPARTMENT_NBR DEPARTMENT_NBR,
a21.CUR_TRN_DT CUR_TRN_DT
from LOOKUP_DAY a21,
LOOKUP_DEPARTMENT a22,
LOOKUP_STORE a23
select a21.MARKET_NBR MARKET_NBR,
max(a24.MARKET_DESC) MARKET_DESC,
sum((a22.COST_AMT * a23.TOT_SLS_DLR)) SUMTSC
from ZZOL00 a21
left outer join COST_STORE_DEP a22
on (a21.DEPARTMENT_NBR = a22.DEPARTMENT_NBR and
a21.CUR_TRN_DT = a22.CUR_TRN_DT and
a21.STORE_NBR = a22.STORE_NBR)
left outer join STORE_DEPARTMENT a23
on (a21.STORE_NBR = a23.STORE_NBR and
a21.DEPARTMENT_NBR = a23.DEPARTMENT_NBR and
a21.CUR_TRN_DT = a23.CUR_TRN_DT),
LOOKUP_MARKET a24
where a21.MARKET_NBR = a24.MARKET_NBR
group by a21.MARKET_NBR
```

SQL 89 Inner Join and SQL 92 Outer & Cross

```
create table ZZOL00 as
select a23.STORE_NBR STORE_NBR,
a23.MARKET_NBR MARKET_NBR,
a22.DEPARTMENT_NBR DEPARTMENT_NBR,
a21.CUR_TRN_DT CUR_TRN_DT
from LOOKUP_DAY a21
cross join LOOKUP_DEPARTMENT a22
cross join LOOKUP_STORE a23
select a21.MARKET_NBR MARKET_NBR,
max(a24.MARKET_DESC) MARKET_DESC,
sum((a22.COST_AMT * a23.TOT_SLS_DLR)) SUMTSC
```

```
from ZZOL00 a21
left outer join COST_STORE_DEP a22
on (a21.DEPARTMENT_NBR = a22.DEPARTMENT_NBR
and
a21.CUR_TRN_DT = a22.CUR_TRN_DT and
a21.STORE_NBR = a22.STORE_NBR)
left outer join STORE_DEPARTMENT a23
on (a21.STORE_NBR = a23.STORE_NBR and
a21.DEPARTMENT_NBR = a23.DEPARTMENT_NBR and
a21.CUR_TRN_DT = a23.CUR_TRN_DT),
LOOKUP_MARKET a24
where a21.MARKET_NBR = a24.MARKET_NBR
group by a21.MARKET_NBR
```

Lookup Table Join Order

Lookup Table Join Order is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This property determines how lookup tables are loaded for being joined. The setting options are

- Partially based on attribute level (behavior prior to version 8.0.1) (default)
- Fully based on attribute level. Lookup tables for lower level attributes are joined before those for higher level attributes

If you select the first option, lookup tables are loaded for join in alphabetic order.

If you select the second option, lookup tables are loaded for join based on attribute levels, and joining is performed on the lowest level attribute first.

Levels at which you can set this

Report, template, and project

Max Tables in Join

Max Tables in Join is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Max Tables in Join property works together with the Max Tables in Join Warning property. It specifies the maximum number of tables in a join. If the maximum number of tables in a join (specified by the Max Tables In Join property) is exceeded, then the Max Tables in Join Warning property decides the course of action.

The table below explains the possible values and their behavior:

Value	Behavior
0	No limit on the number of tables in a join
Number	The maximum number of tables in a join is set to the number specified

Levels at which you can set this

Database instance only

Max Tables in Join Warning

Max Tables in Join Warning is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Max Tables in Join Warning property works in conjunction with the Max Tables in Join property. If the maximum number of tables in a join (specified by the Max Tables in Join property) is exceeded, then this property controls the action taken. The options are to either continue or cancel the execution.

Levels at which you can set this

Database instance, report, and template

Nested Aggregation Outer Joins

The Nested Aggregation Outer Joins VLDB property allows you define when outer joins are performed on metrics that are defined with nested aggregation functions. A nested aggregation function is when one aggregation function is included within another aggregation function. For example, `Sum(Count(Expression))` uses nested aggregation because the Count aggregation is calculated within the Sum aggregation.

These types of metrics can experience unexpected behavior when attempting to use outer joins. This VLDB property provides the following options to control the outer join behavior for metrics that use nested aggregation:

- **Do not perform outer join on nested aggregation** (default): Outer joins are not used for metrics that use nested aggregation, even if the metric is defined to use an outer join. This option reflects the behavior of all pre-9.0 MicroStrategy releases.
- **Do perform outer join on nested aggregation when all formulas have the same level**: If all the inner metrics have the same level, which is lower than the report level, and the formula join type for the outer metric is set to outer, then an outer join is performed on the inner metrics.
- **Do perform downward outer join on nested aggregation when all formulas can downward outer join to a common lower level**: Regardless of whether the inner metrics have the same level, if more than one inner metric has a level which is the child of the levels of other inner metrics and the formula join type for the outer metric is set to outer, then a downward outer join is performed on the relevant inner metrics. The behavior of the downward outer join follows the Downward Outer Join Option VLDB property (see *Nested Aggregation Outer Joins, page 75*).

Levels at which you can set this

Database instance, report, and template

Preserving data using outer joins

For the next two properties, consider the following simple example data.

Store table (lookup)

Store ID	Store Name
1	East
2	Central
3	South
6	North

Fact table

Store ID	Year	Dollar Sales
1	2002	1000
2	2002	2000
3	2002	5000
1	2003	4000
2	2003	6000
3	2003	7000
4	2003	3000
5	2003	1500

The Fact table has data for Store IDs 4 and 5, but the Store table does not have any entry for these two stores. On the other hand, notice that the North Store does not have any entries in the Fact table. This data is used to show examples of how the next two properties work.

Preserve all final pass result elements

For an introduction to this property, see *Preserve all final pass result elements, page 76*. Preserve All Final Pass Result Elements is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Preserve all final pass result elements settings listed below determine how to outer join on the final result and the lookup and relationship tables:

- If you choose the default **Preserve common elements of final pass result table and lookup table** option, the SQL Engine generates an equi-join. Therefore, you only see elements that are common to both tables.
- If you choose the **Preserve all final result pass elements** option, the SQL Engine generates an outer join, and your report contains all of the elements that are in the final result set. When this setting is turned ON, outer joins are generated for any joins from the fact table to the lookup table, as well as to any relationship tables. This is because it is hard to distinguish which table is used as a lookup table and which table is used as a relationship table, the two roles one table often plays. For example, LOOKUP_DAY serves as both a lookup table for the Day attribute, as well as a relationship table for Day and Month.

  This setting should not be used in standard data warehouses, where the lookup tables are properly maintained and all elements in the fact table have entries in the respective lookup table. It should be used only when a certain attribute in the fact table contains more (unique) attribute elements than its corresponding lookup table. For example, in the example above, the Fact Table contains sales for five different stores, but the Store Table contains only four stores. This should not happen in a standard data warehouse because the lookup table, by definition, should contain all the attribute elements. However, this could happen if the fact tables are updated more often than the lookup tables.

- If you choose the **Preserve all elements of final pass result table with respect to lookup table but not relationship table** option, the SQL Engine generates an inner join on all passes except the final pass; on the final pass it generates an outer join.
- If you choose the **Do not listen to per report level setting, preserve elements of final pass according to the setting at attribute level. If this choice is selected at attribute level, it will be treated as preserve common elements (i.e. choice 1)** option at the database instance, report, or template level, the setting for this VLDB property at the attribute level is used. This value should not be selected at the attribute level. If you select this setting at the attribute level, the VLDB property is set to the **Preserve common elements of final pass result table and lookup table** option.

  This setting is useful if you have only a few attributes that require different join types. For example, if among the attributes in a report only one needs to preserve elements from the final pass table, you can set the VLDB property to **Preserve all final pass result elements** setting for that one attribute. You can then set the report to the **Do not listen** setting for the VLDB property. When the report is run, only the attribute set differently causes an outer join in SQL. All other attribute lookup tables will be joined using an equal join, which leads to better SQL performance.

Levels at which you can set this

Database instance, report, template, and attribute

Examples

The first two example results below are based on the *Preserve all final pass result elements, page 76* example above. The third example, for the **Preserve all elements of final pass result table with respect to lookup table but not relationship table** option, is a separate example designed to reflect the increased complexity of that option's behavior.

Example 1: Preserve common elements of final pass result table and lookup table

A report has Store and Dollar Sales on the template.

The "Preserve common elements of final pass result table and lookup table" option returns the following results using the SQL below.

Store	Dollar Sales
East	5000
Central	8000
South	12000

```
select a11.Store_id Store_id,
                max(a12.Store) Store,
                sum(a11.DollarSls) WJXBFS1
from Fact a11
                join Store a12
                  on (a11.Store_id = a12.Store_id)
group by a11.Store_id
```

Example 2: Preserve all final result pass elements

A report has Store and Dollar Sales on the template.

The "Preserve all final result pass elements" option returns the following results using the SQL below. Notice that the data for Store_IDs 4 and 5 are now shown.

Store	Dollar Sales
East	5000
Central	8000
South	12000
	3000
	1500

```
select a11.Store_id Store_id,
                max(a12.Store) Store,
                sum(a11.DollarSls) WJXBFS1
from Fact a11
                left outer join Store a12
                  on (a11.Store_id = a12.Store_id)
```

```
group by a11.Store_id
```

Example 3: Preserve all elements of final pass result table with respect to lookup table but not to relationship table

A report has Country, Metric 1, and Metric 2 on the template. The following fact tables exist for each metric:

CALLCENTER_ID	Fact 1
1	1000
2	2000
1	1000
2	2000
3	1000
4	1000

EMPLOYEE_ID	Fact 2
1	5000
2	6000
1	5000
2	6000
3	5000
4	5000
5	1000

The SQL Engine performs three passes. In the first pass, the SQL Engine calculates metric 1. The SQL Engine inner joins the “Fact Table (Metric 1)” table above with the call center lookup table “LU_CALL_CTR” below:

CALLCENTER_ID	COUNTRY_ID
1	1
2	1
3	2

to create the following metric 1 temporary table, grouped by country, using the SQL that follows:

COUNTRY_ID	Metric 1
1	6000
2	1000

```
create table ZZSP00 nologging as
select a12.COUNTRY_ID  COUNTRY_ID,
                  sum((a11.QTY_SOLD * a11.DISCOUNT))
WJXBFS1
from ORDER_DETAIL a11,
                  LU_CALL_CTR a12
where a11.CALL_CTR_ID = a12.CALL_CTR_ID
group by a12.COUNTRY_ID
```

In the second pass, metric 2 is calculated. The SQL Engine inner joins the "Fact Table (Metric 2)" table above with the employee lookup table "LU_EMPLOYEE" below:

EMPLOYEE_ID	COUNTRY_ID
1	1
2	2
3	2

to create the following metric 2 temporary table, grouped by country, using the SQL that follows:

COUNTRY_ID	Metric 2
1	10000
2	17000

```
create table ZZSP01 nologging as
select a12.COUNTRY_ID  COUNTRY_ID,
                  sum(a11.FREIGHT)  WJXBFS1
from ORDER_FACT a11,
                  LU_EMPLOYEE a12
where a11.EMP_ID = a12.EMP_ID
group by a12.COUNTRY_ID
```

In the third pass, the SQL Engine uses the following country lookup table, "LU_ COUNTRY":

COUNTRY_ID	COUNTRY_DESC
1	United States
3	Europe

The SQL Engine left outer joins the METRIC1_TEMPTABLE above and the LU_COUNTRY table. The SQL Engine then left outer joins the METRIC2_TEMPTABLE above and the LU_COUNTRY table. Finally, the SQL Engine inner joins the results of the third pass to produce the final results.

The "Preserve all elements of final pass result table with respect to lookup table but not to relationship table" option returns the following results using the SQL below.

COUNTRY_ID	COUNTRY_DESC	Metric 1	Metric 2
1	United States	6000	10000
2		1000	17000

```
select pa1.COUNTRY_ID  COUNTRY_ID,
                a11.COUNTRY_NAME  COUNTRY_NAME,
                pa1.WJXBFS1  WJXBFS1,
                pa2.WJXBFS1  WJXBFS2
from ZZSP00 pa1,
                ZZSP01 pa2,
                LU_COUNTRY a11
where pa1.COUNTRY_ID = pa2.COUNTRY_ID and
                pa1.COUNTRY_ID = a11.COUNTRY_ID (+)
```

Preserve all lookup table elements

For an introduction to this property, see *Preserve all lookup table elements, page 81*.

In MicroStrategy 7.1, this property was known as Final Pass Result Table Outer Join to Lookup Table.

The Preserve All Lookup Table Elements property is used to show all attribute elements that exist in the lookup table, even though there is no corresponding fact in the result set. For example, your report contains Store and Sum(Sales), and it is possible that a store does not have any sales at all. However, you want to show all the store names in the final report, even those stores that do not have sales. To do that, you must not rely on the stores in the sales fact table. Instead, you must make sure that all the stores from the lookup table are included in the final report. The SQL Engine needs to do a left outer join from the lookup table to the fact table.

It is possible that there are multiple attributes on the template. To keep all the attribute elements, Analytical Engine needs to do a Cartesian Join between involved attributes' lookup tables before doing a left outer join to the fact table.

Option 1: Preserve common elements of lookup and final pass result table. (default)

The Analytical Engine does a normal (equal) join to the lookup table.

Option 2: Preserve lookup table elements joined to final pass result table based on fact table keys.

Sometimes the fact table level is not the same as the report or template level. For example, a report contains Store, Month, Sum(Sales) metric, but the fact table is at the level of Store, Day, and Item. There are two ways to keep all the store and month elements:

- Do a left outer join first to keep all attribute elements at the Store, Day, and Item level, then aggregate to the Store and Month level.
- Do aggregation first, then do a left outer join to bring in all attribute elements.

  This option is for the first approach. In the example given previously, it makes two SQL passes:

  - **Pass 1**: LOOKUP_STORE cross join LOOKUP_DAY cross join LOOKUP_ITEM ➔TT1
  - **Pass 2**: TT1 left outer join Fact_Table on (store, day, item)

The advantage of this approach is that you can do a left outer join and aggregation in the same pass (pass 2). The disadvantage is that because you do a Cartesian join with the lookup tables at a much lower level (pass 1), the result of the Cartesian joined table (TT1) can be very large.

Option 3: Preserve lookup table elements joined to final pass result table based on template attributes without filter.

This option corresponds to the second approach described above. Still using the same example, it makes three SQL passes:

- **Pass 1**: aggregate the Fact_Table to TT1 at Store and Month. This is actually the final pass of a normal report without turning on this setting.
- **Pass 2**: LOOKUP_STORE cross join LOOKUP_MONTH ➔ TT2
- **Pass 3**: TT2 left outer join TT1 on (store, month)

This approach needs one more pass than the previous option, but the cross join table (TT2) is usually smaller.

Option 4: Preserve lookup table elements joined to final pass result table based on template attributes with filter.

This option is similar to Option 3. The only difference is that the report filter is applied in the final pass (Pass 3). For example, a report contains Store, Month, and Sum(Sales) with a filter of Year = 2002. You want to display every store in every month in 2002,

regardless of whether there are sales. However, you do not want to show any months from other years (only the 12 months in year 2002). Option 4 resolves this issue.

When this VLDB setting is turned ON (Option 2, 3, or 4), it is assumed that you want to keep ALL elements of the attributes in their lookup tables. However, sometimes you want such a setting to affect only some of the attributes on a template. For a report containing Store, Month, Sum(Sales), you may want to show all the store names, even though they have no sales, but not necessarily all the months in the LOOKUP_MONTH table. In 7i, you can individually select attributes on the template that need to preserve elements. This can be done from the **Data** menu, selecting **Report Data Option**, and then choosing **Attribute Join Type**. Notice that the 4 options shown on the upper right are the same as those in the VLDB dialog box (internally they are read from the same location). In the lower-right part, you see individual attributes. By default, all attributes are set to Outer, which means that every attribute participates with the Preserve All Lookup Tables Elements property. You still need to turn on this property to make it take effect, which can be done using either this dialog box or the VLDB dialog box.

Levels at which you can set this

Database instance, report, and template

Examples

The **Preserve common elements of lookup and final pass result table** option simply generates a direct join between the fact table and the lookup table. The results and SQL are as follows.

Store	Dollar Sales
East	5000
Central	8000
South	12000

```
select a11.Store_id Store_id,
                max(a12.Store) Store,
                sum(a11.DollarSls) WJXBFS1
from Fact a11
                join Store a12
                  on (a11.Store_id = a12.Store_id)
group by a11.Store_id
```

The “Preserve lookup table elements joined to final pass result table based on fact keys” option creates a temp table that is a Cartesian join of all lookup table key columns. Then the fact table is outer joined to the temp table. This preserves all lookup table elements. The results and SQL are as below:

Store	Dollar Sales
East	5000
Central	8000
South	12000
North	

```
select distinct a11.Year Year
into #ZZOL00
from Fact a11
select pa1.Year Year,
                a11.Store_id Store_id
into #ZZOL01
from #ZZOL00 pa1
                cross join Store a11
select pa2.Store_id Store_id,
                max(a12.Store) Store,
                sum(a11.DollarSls) WJXBFS1
from #ZZOL01 pa2
                left outer join Fact a11
                  on (pa2.Store_id = a11.Store_id and
                pa2.Year = a11.Year)
                join Store a12
                  on (pa2.Store_id = a12.Store_id)
group by pa2.Store_id
drop table #ZZOL00
drop table #ZZOL01
```

The “Preserve lookup table elements joined to final pass result table based on template attributes without filter” option preserves the lookup table elements by left outer joining to the final pass of SQL and only joins on attributes that are on the template. For this example and the next, the filter of “Store not equal to Central” is added. The results and SQL are as follows:

Store	Dollar Sales
East	5000
Central	
South	12000
North	

```
select a11.Store_id Store_id,
                sum(a11.DollarSls) WJXBFS1
into #ZZT5X00003UOL000
from Fact a11
where a11.Store_id not in (2)
group by a11.Store_id
select a11.Store_id Store_id,
                a11.Store Store,
                pa1.WJXBFS1 WJXBFS1
from Store a11
                left outer join #ZZT5X00003UOL000 pa1
                  on (a11.Store_id = pa1.Store_id)
```

```
drop table #ZZT5X00003UOL000
```

The "Preserve lookup table elements joined to final pass result table based on template attributes with filter" option is the newest option and is the same as above, but you get the filter in the final pass. The results and SQL are as follows:

Store	Dollar Sales
East	5000
South	12000
North	

```
select a11.Store_id Store_id,
                sum(a11.DollarSls) WJXBFS1
into #ZZT5X00003XOL000
from Fact a11
where a11.Store_id not in (2)
group by a11.Store_id
select a11.Store_id Store_id,
                a11.Store Store,
                pa1.WJXBFS1 WJXBFS1
from Store a11
                left outer join #ZZT5X00003XOL000 pa1
                  on (a11.Store_id = pa1.Store_id)
where a11.Store_id not in (2)
drop table #ZZT5X00003XOL000
```

Modifying third-party cube sources in MicroStrategy: MDX

The table below summarizes the MultiDimensional Expression (MDX) related VLDB properties. These properties apply only to MDX cube reports using data from an MDX cube. MDX cubes are also referred to as MDX cube sources. MicroStrategy supports reporting and analysis with SAP BW, Microsoft Analysis Services, Hyperion Essbase, and IBM Cognos TM1. Additional details about each property, including examples where necessary, are provided in the sections following the table.

In the table below, the default values for each VLDB property are the general defaults that can be applied most broadly for the set of certified MDX cube sources. Certain VLDB properties use different default settings depending on which MDX cube source you are using. To determine all default VLDB property settings for the MDX cube source you are reporting on, follow the steps provided in *Default VLDB settings for specific data sources, page 225*.

Property	Description	Possible Values	Default Value
Format for Date/Time Values Coming from Data Source	Defines the date format used in your MDX cube source. This ensures the date data is integrated into MicroStrategy	User-defined	DD.MM.YYYY

Property	Description	Possible Values	Default Value
	correctly.		
MDX Add Fake Measure	Determines how MDX cube reports that only include attributes are processed in order to improve performance in certain scenarios.	• Do not add a fake measure to an attribute-only MDX report • Add a fake measure to an attribute-only MDX report	Add a fake measure to an attribute-only MDX report
MDX Add Non Empty	Determines whether or not data is returned from rows that have null values.	• Do not add the non-empty keyword in the MDX select clause • Add the non-empty keyword in the MDX select clause only if there are metrics on the report • Always add the non-empty keyword in the MDX select clause	Add the non-empty keyword in the MDX select clause only if there are metrics on the report
MDX Cell Formatting	Defines whether the metric values in MicroStrategy MDX cube reports inherit their value formatting from an MDX cube source.	• MDX metric values are formatted per column • MDX metric values are formatted per cell	MDX metric values are formatted per column
MDX Has Measure Values In Other Hierarchies	Determines how null values are identified if you use the *Modifying third-party cube sources in MicroStrategy: MDX* VLDB property to ignore null values coming from MDX cube sources.	• Only include the affected hierarchy in the "has measure values" set definition • Include all template hierarchies in the "has measure values" set definition	Only include the affected hierarchy in the "has measure values" set definition
MDX Level Number Calculation Method	Determines whether level (from the bottom of the hierarchy up) or generation (from the top of the hierarchy down) should be used to populate the report results.	• Use actual level number • Use generation number to calculate level number	Use actual level number
MDX Measure Values to Treat as Null	Allows you to specify what measure values are defined as NULL values, which can help	User-defined	X

Property	Description	Possible Values	Default Value
	to support how your SAP environment handles non-calculated measures.		
MDX Non Empty Optimization	Determines how null values from an MDX cube source are ignored using the non-empty keyword when attributes from different hierarchies (dimensions) are included on the same MDX cube report.	• No non-empty optimization • Non-empty optimization, use default measure • Non-empty optimization, use first measure on template • Non-empty optimization, use all measures on template	No non-empty optimization
MDX Remember Measure Dimension Name	Defines how the name of the measure dimension is determined for an MDX cube source.	• Do not remember the name of the measure dimension • Remember the name of the measure dimension • Read the name of the measure dimension from the “Name of Measure Dimension” VLDB setting	Do not remember the name of the measure dimension
MDX TopCount Support	Determines whether TopCount is used in place of Rank and Order to support certain MicroStrategy features such as metric filter qualifications.	• Do not use TopCount in the place of Rank and Order • Use TopCount instead of Rank and Order	Use TopCount instead of Rank and Order
Modifying third-party cube sources in MicroStrategy: MDX	Determines how date qualifications are processed for MDX cube sources.	• Do not treat a date qualification on a key form as a date qualification on an ID form: • Treat a date qualification on a key form as a date qualification on an ID form	Treat a date qualification on a key form as a date qualification on an ID form

Property	Description	Possible Values	Default Value
MDX Verify Limit Filter Literal Level	Supports an MDX cube reporting scenario in which filters are created on attribute ID forms and metrics.	• Do not verify the level of literals in limit or filter expressions • Verify the level of literals in limit or filter expressions	Do not verify the level of literals in limit or filter expressions
Modifying third-party cube sources in MicroStrategy: MDX	Defines the name of the measures dimension in an MDX cube source.	User-defined	[Measures]

Format for Date/Time Values Coming from Data Source

Date data can be stored in a variety of formats in MDX cube sources. To ensure that your date data from your MDX cube source is integrated into MicroStrategy with the correct format, you can use the Format for Date/Time Values Coming from Data Source VLDB property to define the date format used in your MDX cube source.

The default date format is DD.MM.YYYY. For example, the date of July 4, 1776 is represented as 04.07.1776.

For information on supporting MDX cube source date data in MicroStrategy, see the *MicroStrategy MDX Cube Reporting Guide*.

Levels at which you can set this

Database instance only

MDX Add Fake Measure

MDX Add Fake Measure is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

It is a common practice to include both attributes and metrics on an MDX cube report. However, MDX cube reports can also contain only attributes to review attribute information. If this type of MDX cube report accesses data that is partitioned within the MDX cube source, the report can require additional resources and impact the performance of the report. To avoid this performance issue, the MDX Add Fake Measure VLDB property provides the following options:

- **Do not add fake measure to attribute-only MDX report**: MDX cube reports that only contain attributes without any metrics are processed as normal. This can cause additional processing to be required for this type of MDX cube report if it accesses data that is partitioned within the MDX cube source. This is the default option for SAP and Essbase MDX cube sources.

- **Add a fake measure to an attribute-only MDX report**: MDX cube reports that only contain attributes without any metrics also include an additional structure that acts as a metric, although no metrics are displayed on the report. This can improve performance of MDX cube reports that only contain attributes and also access data that is partitioned within the MDX cube source. This is the default option for Microsoft Analysis Services MDX cube sources.

Levels at which you can set this

Database instance and report

MDX Add Non Empty

MDX Add Non Empty is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property determines how null values can be returned to MicroStrategy from an MDX cube source and displayed on MDX cube reports. To determine whether null data should be displayed on MDX cube reports when attributes from different hierarchies (dimensions) are included on the same MDX cube report, see *MDX Add Non Empty, page 89*.

You can choose from the following settings:

- **Do not add the non-empty keyword in the MDX select clause**: When this option is selected, data is returned from rows that contain data and rows that have null metric values (similar to an outer join in SQL). The null values are displayed on the MDX cube report.
- **Add the non-empty keyword in the MDX select clause only if there are metrics on the report** (default): When this option is selected, and metrics are included on an MDX cube report, data is not returned from the MDX cube source when the default metric in the MDX cube source has null data. Any data not returned is not included on MDX cube reports (similar to an inner join in SQL). If no metrics are present on an MDX cube report, then all values for the attributes are returned and displayed on the MDX cube report.
- **Always add the non-empty keyword in the MDX select clause**: When this option is selected, data is not returned from the MDX cube source when a metric on the MDX cube report has null data. Any data not returned is not included on MDX cube reports (similar to an inner join in SQL).

Levels at which you can set this

Database instance and report

For more information on MDX cube sources, see the MDX Cube Reporting Guide.

Example

Do not add the non-empty keyword in the MDX select clause

```
with set [dim0_select_members] as '{[0D_SOLD_TO].
[LEVEL01].members}'
set [dim1_select_members] as '{[0CALQUARTER].
[LEVEL01].members}'
select {[Measures].[3STVV9JH7ATAV9YJN06S7ZKSQ]} on
columns,CROSSJOIN(hierarchize({[dim0_select_members]}),
hierarchize({[dim1_select_members]})) dimension
properties [0D_SOLD_TO].[20D_SOLD_TO], [0D_SOLD_TO].[10D_
SOLD_TO] on rows
from [0D_DECU/QCUBE2]
```

Add the non-empty keyword in the MDX select clause

```
with set [dim0_select_members] as '{[0D_SOLD_TO].
[LEVEL01].members}'set [dim1_select_members] as '
{[0CALQUARTER].[LEVEL01].members}'
select {[Measures].[3STVV9JH7ATAV9YJN06S7ZKSQ]} on
columns,
non empty CROSSJOIN(hierarchize({[dim0_select_members]}),
hierarchize({[dim1_select_members]})) dimension
properties [0D_SOLD_TO].[20D_SOLD_TO], [0D_SOLD_TO].[10D_
SOLD_TO] on rowsfrom [0D_DECU/QCUBE2]
```

MDX Cell Formatting

With the MDX Cell Formatting VLDB property, you can specify for the metric values in MicroStrategy MDX cube reports to inherit their value formatting from an MDX cube source. This enables MicroStrategy MDX cube reports to use the same data formatting available in your MDX cube source. It also maintains a consistent view of your MDX cube source data in MicroStrategy.

Inheriting value formats from your MDX cube source also enables you to apply multiple value formats to a single MicroStrategy metric.

This VLDB property has the following options:

- **MDX metric values are formatted per column** (default): If you select this option, MDX cube source formatting is not inherited. You can only apply a single format to all metric values on an MDX cube report.
- **MDX metric values are formatted per cell**: If you select this option, MDX cube source formatting is inherited. Metric value formats are determined by the formatting that is available in the MDX cube source, and metric values can have different formats.

For examples of using these options and steps to configure your MDX cube sources properly, see the *MDX Cube Reporting Guide*.

Levels at which you can set this

Database instance and report

MDX Has Measure Values In Other Hierarchies

MDX Has Measure Values In Other Hierarchies is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property determines how null values are identified if you use the MDX Non Empty Optimization VLDB property (see *MDX Has Measure Values In Other Hierarchies, page 91*) to ignore null values coming from MDX cube sources.

If you define the MDX Non Empty Optimization VLDB property as No non-empty optimization, then this VLDB property has no effect on how null values are ignored. If you use any other option for the MDX Non Empty Optimization VLDB property, you can choose from the following settings:

- **Only include the affected hierarchy in the "has measure values" set definition**: Only a single hierarchy on the MDX cube report is considered when identifying and ignoring null values. This requires fewer resources to determine the null values, but some values can be mistakenly identified as null values in scenarios such as using calculated members in an MDX cube source.
- **Include all template hierarchies in the "has measure values" set definition**: All hierarchies that are part of an MDX cube report are considered when identifying and ignoring null values. This can help to ensure that some values are not lost when MicroStrategy ignores null values from the MDX cube source. Including all hierarchies to identify null values can require additional system resources and time to complete.

Levels at which you can set this

Database instance and report

MDX Level Number Calculation Method

MDX Level Number Calculation is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property is useful only for MDX cube reports that access an Oracle Hyperion Essbase MDX cube source. To help illustrate the functionality of the property, consider an unbalanced hierarchy with the levels Products, Department, Category, SubCategory, Item, and SubItem. The image below shows how this hierarchy is populated on a report in MicroStrategy.

Products.Levels(5)	Products.Levels(4)	Products.Levels(3)	Products.Levels(2)	Products.Levels(1)	Products.Levels(0)
Products		Dept1	Cat1	SubCat1	Item1
					Item2
				SubCat2	Item3
					Item4
			Cat2	SubCat3	Item5
					Item6
				SubCat4	Item7
					Item8
	Dept2		Cat3	SubCat5	Item9
				SubCat6	Item11
					Item12
		Cat4		SubCat7	Item13
					Item14
			SubCat8		Item15
				Item16	SubItem1
					SubItem2

The level SubItem causes the hierarchy to be unbalanced, which displaces the levels of the hierarchy when populated on a report in MicroStrategy. For more information on unbalanced and ragged hierarchies, see the *MDX Cube Reporting Guide*.

You can choose from the following settings:

- **Use actual level number** (default): When this option is selected, an unbalanced or ragged hierarchy from Essbase is populated on a grid from the bottom of the hierarchy up, as shown in the image above.
- **Use generation number to calculate level number:** When this option is selected, an unbalanced or ragged hierarchy from Essbase is populated on a grid from the top of the hierarchy down. If this setting is selected for the example scenario described above, the report is populated as shown in the image below.

Products.Levels(5)	Products.Levels(4)	Products.Levels(3)	Products.Levels(2)	Products.Levels(1)	Products.Levels(0)
Products	Dept1	Cat1	SubCat1	Item1	
				Item2	
			SubCat2	Item3	
				Item4	
		Cat2	SubCat3	Item5	
				Item6	
			SubCat4	Item7	
				Item8	
	Dept2	Cat3	SubCat5	Item9	
			SubCat6	Item11	
				Item12	
		Cat4	SubCat7	Item13	
				Item14	
			SubCat8	Item15	
				Item16	SubItem1
					SubItem2

The unbalanced hierarchy is now displayed on the report with an accurate representation of the corresponding levels.

Setting this VLDB property to Add the generation number property for a ragged hierarchy from Essbase can cause incorrect formatting.

Levels at which you can set this

Database instance and report

MDX Measure Values to Treat as Null

MDX Measure Values to Treat as Null is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property allows you to specify what measure values are defined as NULL values, which can help to support how your SAP environment handles non-calculated measures. The default value to treat as NULL is X. This supports defining non-calculated measures as NULL values for SAP 7.4 environments.

Levels at which you can set this

Database instance and report

MDX Non Empty Optimization

MDX Non Empty Optimization is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property determines how null values from an MDX cube source are ignored using the non-empty keyword when attributes from different hierarchies (dimensions) are included on the same MDX cube report.

You can choose from the following settings:

- **No non-empty optimization** (default): The non-empty keyword is not included during the cross join of data. By selecting this option, all null data is included on the MDX cube report. Including all null data can require more system resources to perform the necessary cross joins.
- **Non-empty optimization, use default measure**: The non-empty keyword is added to any required cross joins based on the default measure within the MDX cube source. Data is only displayed on an MDX cube report for rows in which the default measure within the MDX cube source has data. If you use this option, you can also control whether null values from MDX cube sources are ignored using the VLDB property MDX Has Measure Values In Other Hierarchies (see *MDX Non Empty Optimization, page 93*).
- **Non-empty optimization, use first measure on template**: The non-empty keyword is added to any required cross joins based on the first metric used on an MDX cube report. Data is only displayed on an MDX cube report for rows in which the first metric used on an MDX cube report has data. For example, if Revenue and Profit metrics are on an MDX cube report and Revenue is in the first column (left-most column), the non-empty keyword is added based on the Revenue metric. In this scenario, null or empty data may still be returned for the Profit metric. If you use this option, you can also control whether null values from MDX cube sources are ignored using the VLDB property MDX Has Measure Values In Other Hierarchies (see *MDX Non Empty Optimization, page 93*).
- **Non-empty optimization, use all measures on template**: The non-empty keyword is added to any required cross joins based on all metrics used on an MDX

cube report. Data is only displayed on an MDX cube report for rows in which at least one of the metrics used on an MDX cube report has data. For example, Revenue and Profit metrics are on an MDX cube report, which includes the following data:

Year	Category	Revenue	Profit
2008	Books	$1,000,000	$300,000
	Electronics	$2,500,000	
	Movies		$500,000
	Music		

By selecting this option, the following data would be returned on the MDX cube report:

Year	Category	Revenue	Profit
2008	Books	$1,000,000	$300,000
	Electronics	$2,500,000	
	Movies		$500,000

The row for Music is not displayed because all the metrics have null values. If you use this option, you can also control whether null values from MDX cube sources are ignored using the VLDB property MDX Has Measure Values In Other Hierarchies (see *MDX Non Empty Optimization, page 93*).

Levels at which you can set this

Database instance and report

MDX Remember Measure Dimension Name

MDX Remember Measure Dimension Name is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property defines how the name of the measure dimension is determined for an MDX cube source. You can choose from the following settings:

- **Do not remember the name of the measure dimension**: The MDX cube source is not analyzed to determine the name of the measure dimension. Since most MDX cube sources use `[Measures]` as the measure dimension name and MicroStrategy recognizes this default name, this option is recommended for most MDX cube sources.
- **Remember the name of the measure dimension**: The MDX cube source is analyzed to determine the name of the measure dimension. The name returned is

then used later when querying the MDX cube source. This option can be used when an MDX cube source does not use `[Measures]` as the measure dimension name, which is the default used for most MDX cube sources. Essbase is the MDX cube source that most commonly uses a measure dimension name other than `[Measures]`.

- **Read the name of the measure dimension from the "Name of Measure Dimension" VLDB setting**: The measure dimension name defined using the Name of Measure Dimension VLDB property (see *MDX Remember Measure Dimension Name, page 94*) is used as the measure dimension name. You can use this option if the MDX cube source does not use `[Measures]` as the measure dimension name, and you know what alternative name is used for the measure dimension.

Levels at which you can set this

Database instance only

MDX TopCount Support

MDX TopCount Support is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property determines whether TopCount is used in place of Rank and Order to support certain MicroStrategy features such as metric filter qualifications. TopCount can be used with SAP BW and Microsoft Analysis Services MDX cube sources.

You can choose from the following settings:

- **Do not use TopCount in the place of Rank and Order**: The functions Rank and Order are always used instead of TopCount. This option supports backwards compatibility.
- **Use TopCount instead of Rank and Order** (default): The function TopCount is automatically used in place of Rank and Order when necessary to support certain MicroStrategy features. This includes scenarios such as using metric filter qualifications on MDX cube reports.

Levels at which you can set this

Database instance and report

MDX Treat Key Date Qualification As ID Date Qualification

MDX Treat Key Date Qualification As ID Date Qualification is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property determines how date qualifications are processed for MDX cube sources. You can choose from the following settings:

- **Do not treat a date qualification on a key form as a date qualification on an ID form**: This option processes date qualifications by using the member properties. While this can impact performance, you can use this option to support date qualifications on data that cannot be processed by using the unique name.
- **Treat a date qualification on a key form as a date qualification on an ID form** (default): This option provides the best performance for processing date qualifications by using the unique name rather than the member properties.

Levels at which you can set this

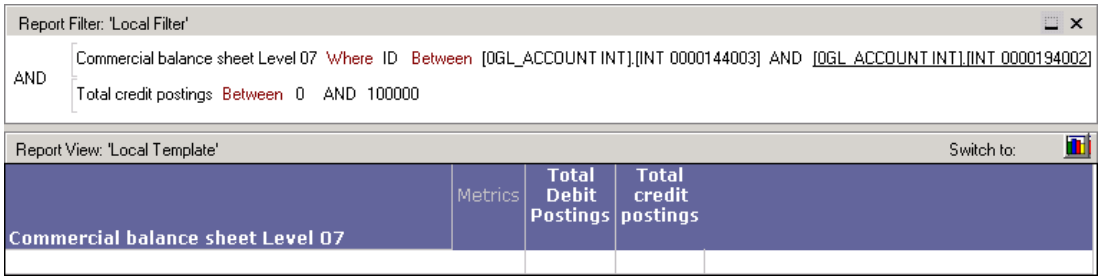
Database instance and report

MDX Verify Limit Filter Literal Level

MDX Verify Limit Filter Literal Level is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property supports a unique scenario when analyzing MDX cube reports. An example of this scenario is provided below.

You have an MDX cube report that includes a low level attribute on the report, along with some metrics. You create a filter on the attribute's ID form, where the ID is between two ID values. You also include a filter on a metric. Below is an example of such an MDX cube report definition:



When you run the report, you receive an error that alerts you that an unexpected level was found in the result. This is because the filter on the attribute's ID form can include other levels due to the structure of ID values in some MDX cube sources. When these other levels are included, the metric filter cannot be evaluated correctly by default.

You can support this type of report by modifying the MDX Verify Limit Filter Literal Level. This VLDB property has the following options:

- **Do not verify the level of literals in limit or filter expressions** (default): While the majority of MDX cube reports execute correctly when this option is selected, the scenario described above will fail.
- **Verify the level of literals in limit or filter expressions**: Selecting this option for an MDX cube report allows reports fitting the scenario described above to execute correctly. This is achieved by adding an intersection in the MDX statement to support the execution of such an MDX cube report. For example, the MDX cube

report described in the scenario above executes correctly and displays the following data.

Commercial balance sheet Level 07	Metrics	Total Debit Postings	Total credit postings
Receivables CC 3000		26,863.28 EUR	26,863.28 EUR
Clearing with 2000		41,092.39 EUR	0.00 EUR

Levels at which you can set this

Database instance and report

Name of Measure Dimension

Name of Measure Dimension is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This VLDB property defines the name of the measures dimension in an MDX cube source. The default name for the measures dimension is `[Measures]`. If your MDX cube source uses a different name for the measures dimension, you must modify this VLDB property to match the name used in your MDX cube source. Requiring this change is most common when connecting to Essbase MDX cube sources, which do not always use `[Measures]` as the measure dimension name.

Identifying the name of the measure dimension is also configured using the MDX Remember Measure Dimension Name VLDB property, as described in *Name of Measure Dimension, page 97*.

Levels at which you can set this

Database instance and report

Calculating data: Metrics

The table below summarizes the Metrics VLDB properties. Additional details about each property, including examples where necessary, are provided in the sections following the table.

Property	Description	Possible Values	Default Value
Absolute Non-Agg Metric Query Type	The Analytical Engine can either: • Perform the non-aggregation calculation with a subquery, or	• Use subquery • Use temp table as set in the Fallback Table Type setting	Use subquery

Property	Description	Possible Values	Default Value
	• Place the results that would have been selected from a subquery into an intermediate table and join that table to the rest of the query.		
Compute Non-Agg before/after OLAP Functions (e.g. Rank) Calculated in Analytical Engine	This property controls whether the non-aggregation calculation is performed before or after an Analytical Engine calculation. Use this property to determine, for example, whether the engine ranks the stores and then performs a non-aggregation calculation, or performs the non-aggregation calculation first.	• Calculate non-aggregation before OLAP Functions/Rank • Calculate non-aggregation after OLAP Functions/Rank	Calculate non-aggregation before OLAP Functions/Rank
Count Compound Attribute	Compound attributes are usually counted by concatenating the keys of all the attributes that form the key. If the database platform does not support COUNT on concatenated strings, this property should be disabled.	• COUNT expression enabled • COUNT expression disabled	COUNT expression enabled
COUNT(column) Support	Some database platforms do not support count on a column (COUNT(COL)). This property converts the COUNT (COL) statement to a COUNT (*).	• Use COUNT(column) • Use COUNT(*)	Use COUNT(column)
Default to Metric Name	Allows you to choose whether you want to use the metric name as the column alias or whether to use a MicroStrategy- generated name.	• Do not use the metric name as the default metric column alias • Use the metric name as the default metric column alias	Do not use the metric name as the default metric column alias
Integer Constant in Metric	This property determines whether to add a ".0" after the integer.	• Add ".0" to integer constant in metric expression • Do Not Add ".0" to integer constant in metric expression	Add ".0"' to integer constant in metric expression
Join Across Datasets		• Disallow joins based on unrelated common attributes	Disallow joins based on unrelated common attributes

Property	Description	Possible Values	Default Value
		• Allow joins based on unrelated common attributes:	
Max Metric Alias Size	Maximum size of the metric alias string	User-defined	256
Metric Join Type	Type of join used in a metric.	• Inner Join • Outer Join	Inner Join
Non-Agg Metric Optimization	Influences the behavior for non-aggregation metrics by either optimizing for smaller temporary tables or for less fact table access.	• Optimized for less fact table access • Optimized for smaller temp table	Optimized for less fact table access
NULL Check	Indicates how to handle arithmetic operations with NULL values.	• Do nothing • Check for NULL in all queries • Check for NULL in temp table join only	Check for NULL in temp table join only
Separate COUNT DISTINCT	Indicates how to handle COUNT (and other aggregation functions) when DISTINCT is present in the SQL.	• One pass • Multiple count distinct, but count expression must be the same • Multiple count distinct, but only one count distinct per pass • No count distinct, use select distinct and count(*) instead	No count distinct, use select distinct and count(*) instead
Smart Metric Transformation	Determines the evaluation order to support variance and variance percentage transformations on smart metric or compound metric results.	• False • True	False

Property	Description	Possible Values	Default Value
Subtotal Dimensionality Use	Determines how the level of calculation is defined for metrics that are included on reports that utilize the OLAP Services feature dynamic aggregation.	• Use only the grouping property of a level metric for dynamic aggregation (default): • Use only the grouping property of a level subtotal for dynamic aggregation: • Use both the grouping and filtering property of a level metric for dynamic aggregation: • Use both the grouping and filtering property of a level subtotal for dynamic aggregation:	Use only the grouping property of a level metric for dynamic aggregation
Transformable AggMetric	Define metrics that should be used to perform transformations on compound metrics that use nested aggregation.	• False • True	False
Transformation Role Processing	Indicates how to handle the transformation dates calculation.	• 7.1 style. Apply transformation to all applicable attributes • 7.2 style. Only apply transformation to highest common child when it is applicable to multiple attributes	7.1 style. Apply transformations to all applicable attributes
Zero Check	Indicates how to handle division by zero.	• Do nothing • Check for zero in all queries • Check for zero in temp table join only	Check for zero in all queries

Absolute Non-Agg Metric Query Type

Absolute Non-Agg Metric Query Type is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

When a report contains an absolute non-aggregation metric, the pass that gets the non-aggregated data can be performed in a subquery or in a temporary table.

- **Use Temp Table as set in the Fallback Table Type setting**: When this option is set, the table creation type follows the option selected in the VLDB property Fallback Table Type. The SQL Engine reads the Fallback Table Type VLDB setting and determines whether to create the intermediate table as a true temporary table or a permanent table.

  In most cases, the default Fallback Table Type VLDB setting is Temporary table. However, for a few databases, like UDB for 390, this option is set to Permanent table. These databases have their Intermediate Table Type defaulting to True Temporary Table, so you set their Fallback Table Type to Permanent. If you see permanent table creation and you want the absolute non-aggregation metric to use a True Temporary table, set the Fallback Table Type to Temporary table on the report as well.

- **Use subquery** (default): With this setting, the engine performs the non-aggregation calculation with a subquery.

Levels at which you can set this

Database instance, report, and template

Examples

Use Sub-query

```
select a11.CLASS_NBR CLASS_NBR,
a12.CLASS_DESC CLASS_DESC,
sum(a11.TOT_SLS_QTY) WJXBFS1
from DSSADMIN.MARKET_CLASS a11,
DSSADMIN.LOOKUP_CLASS a12
where a11.CLASS_NBR = a12.CLASS_NBR
and (((a11.MARKET_NBR)
in (select s21.MARKET_NBR
from DSSADMIN.LOOKUP_STORE s21
where s21.STORE_NBR in (3, 2, 1)))
and ((a11.MARKET_NBR)
in (select min(c11.MARKET_NBR)
from DSSADMIN.LOOKUP_MARKET c11
where ((c11.MARKET_NBR)
in (select s21.MARKET_NBR
from DSSADMIN.LOOKUP_STORE s21
where s21.STORE_NBR in (3, 2, 1))))))
group by a11.CLASS_NBR,
a12.CLASS_DESC
```

Use Temporary Table as set in the Fallback Table Type setting

```
create table TPZZOP00  as
select min(c11.MARKET_NBR)  WJXBFS1
from DSSADMIN.LOOKUP_MARKET c11
where ((c11.MARKET_NBR)
 in (select s21.MARKET_NBR
 from DSSADMIN.LOOKUP_STORE s21
 where s21.STORE_NBR in (3, 2, 1)))
select a11.CLASS_NBR  CLASS_NBR,
 a12.CLASS_DESC  CLASS_DESC,
```

```
 sum(a11.TOT_SLS_QTY)  WJXBFS1
from DSSADMIN.MARKET_CLASS a11,
 TPZZOP00 pa1,
 DSSADMIN.LOOKUP_CLASS a12
where a11.MARKET_NBR = pa1.WJXBFS1 and
 a11.CLASS_NBR = a12.CLASS_NBR
 and ((a11.MARKET_NBR)
 in (select s21.MARKET_NBR
 from DSSADMIN.LOOKUP_STORE s21
 where s21.STORE_NBR in (3, 2, 1)))
group by a11.CLASS_NBR,
 a12.CLASS_DESC
```

Compute Non-Agg before/after OLAP Functions (e.g. Rank) Calculated in Analytical Engine

Compute Non-Agg Before/After OLAP Functions/Rank is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

When reports contain calculations based on non-aggregation metrics, this property controls the order in which the non-aggregation and calculations are computed.

Levels at which you can set this

Database instance, report, and template

Examples

Calculate Non-Aggregation Before Analytical (default)

```
select a12.YEAR_ID YEAR_ID,
                sum(a11.TOT_SLS_QTY) WJXBFS1
from HARI_REGION_DIVISION a11
                join HARI_LOOKUP_DAY a12
                 on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
where a11.CUR_TRN_DT)
 in (select min(a11.CUR_TRN_DT)
                from HARI_LOOKUP_DAY a11
                group by a11.YEAR_ID))
group by a12.YEAR_ID
create table #ZZTIS00H5J7MQ000(
                YEAR_ID DECIMAL(10, 0))
[Placeholder for an analytical SQL]
select a12.YEAR_ID YEAR_ID,
                max(a13.YEAR_DESC) YEAR_DESC,
                sum(a11.TOT_SLS_QTY) TSQDIMYEARNA
from HARI_REGION_DIVISION a11
                join HARI_LOOKUP_DAY a12
                  on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
                join #ZZTIS00H5J7MQ000 pa1
                  on (a12.YEAR_ID = pa1.YEAR_ID)
                join HARI_LOOKUP_YEAR a13
                  on (a12.YEAR_ID = a13.YEAR_ID)
where ((a11.CUR_TRN_DT)
 in (select min(a15.CUR_TRN_DT)
                from #ZZTIS00H5J7MQ000 pa1
                        join HARI_LOOKUP_DAY a15
                          on (pa1.YEAR_ID = a15.YEAR_ID)
```

```
                group by pa1.YEAR_ID))
group by a12.YEAR_ID
```

Calculate Non-Aggregation After Analytical

```
select a11.CUR_TRN_DT CUR_TRN_DT,
                a12.YEAR_ID YEAR_ID,
                sum(a11.TOT_SLS_QTY) WJXBFS1
from HARI_REGION_DIVISION a11
                join HARI_LOOKUP_DAY a12
                  on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
group by a11.CUR_TRN_DT,
                a12.YEAR_ID
create table #ZZTIS00H5J8NB000(
                CUR_TRN_DT DATETIME,
                YEAR_ID DECIMAL(10, 0),
                WJXBFS1 FLOAT)
[Placeholder for an analytical SQL]
insert into #ZZTIS00H5J8NB000 values (CONVERT
(datetime, '1993-12-01 00:00:00', 120), 1993,
44)
```

[The rest of the INSERT statements have been omitted from display].

```
select distinct pa1.YEAR_ID YEAR_ID,
                pa1.WJXBFS1 WJXBFS1
from #ZZTIS00H5J8NB000 pa1
where ((pa1.CUR_TRN_DT)
 in (select min(c11.CUR_TRN_DT)
                from HARI_LOOKUP_DAY c11
                group by c11.YEAR_ID))
create table #ZZTIS00H5J8MQ001(
                YEAR_ID DECIMAL(10, 0),
                WJXBFS1 FLOAT)
[Placeholder for an analytical SQL]
select a12.YEAR_ID YEAR_ID,
                max(a13.YEAR_DESC) YEAR_DESC,
                sum(a11.TOT_SLS_QTY) TSQDIMYEARNA
from HARI_REGION_DIVISION a11
                join HARI_LOOKUP_DAY a12
                  on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
                join #ZZTIS00H5J8MQ001 pa2
                  on (a12.YEAR_ID = pa2.YEAR_ID)
                join HARI_LOOKUP_YEAR a13
                  on (a12.YEAR_ID = a13.YEAR_ID)
where ((a11.CUR_TRN_DT)
 in (select min(a15.CUR_TRN_DT)
                from #ZZTIS00H5J8MQ001 pa2
                        join HARI_LOOKUP_DAY a15
                          on (pa2.YEAR_ID = a15.YEAR_ID)
                group by pa2.YEAR_ID))
group by a12.YEAR_ID
```

Count Compound Attribute

Count Compound Attribute is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Compound attributes are usually counted by concatenating the keys of all of the attributes that form the key.

If your database platform does not support COUNT on concatenated strings, the Count Compound Attribute property should be disabled.

Levels at which you can set this

Database instance only

Examples

COUNT expression enabled (default)

```
select a21.DIVISION_NBR DIVISION_NBR,
max(a22.DIVISION_DESC) DIVISION_DESC,
count(distinct char(a21.ITEM_NBR) || ||
char(a21.CLASS_NBR)) ITEM_COUNT
from LOOKUP_ITEM a21
join LOOKUP_DIVISION a22
on (a21.DIVISION_NBR = a22.DIVISION_NBR)
group by a21.DIVISION_NBR
```

COUNT expression disabled

```
create table TEMP1 as
select distinct a21.DIVISION_NBR DIVISION_NBR,
a21.ITEM_NBR ITEM_NBR,
a21.CLASS_NBR CLASS_NBR
from LOOKUP_ITEM a21
select a22.DIVISION_NBR DIVISION_NBR,
max(a22.DIVISION_DESC) DIVISION_DESC,
count(a21.ITEM_NBR) ITEM_COUNT
from TEMP1 a21
join LOOKUP_DIVISION a22
on (a21.DIVISION_NBR = a22.DIVISION_NBR)
group by a22.DIVISION_NBR
```

COUNT(column) Support

COUNT(column) Support is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The COUNT(column) Support property is used to specify whether COUNT on a column is supported or not. If it is not supported, the COUNT(column) is computed by using intermediate tables and COUNT(*).

Levels at which you can set this

Database instance only

Examples

Use COUNT(column)

```
select a11.STORE_NBR STORE_NBR,
                max(a12.STORE_DESC) STORE_DESC,
```

```
                count(distinct a11.COST_AMT) COUNTDISTINCT
from HARI_COST_STORE_DEP a11
                join HARI_LOOKUP_STORE a12
                  on (a11.STORE_NBR = a12.STORE_NBR)
group by a11.STORE_NBR
```

Use COUNT(*)

```
select a11.STORE_NBR STORE_NBR,
                a11.COST_AMT WJXBFS1
into #ZZTIS00H5JWDA000
from HARI_COST_STORE_DEP a11
select distinct pa1.STORE_NBR STORE_NBR,
                pa1.WJXBFS1 WJXBFS1
into #ZZTIS00H5JWOT001
from #ZZTIS00H5JWDA000 pa1
where pa1.WJXBFS1 is not null
select           pa2.STORE_NBR STORE_NBR,
                max(a11.STORE_DESC) STORE_DESC,
                count(*) WJXBFS1
from            #ZZTIS00H5JWOT001 pa2
                join HARI_LOOKUP_STORE a11
                  on (pa2.STORE_NBR = a11.STORE_NBR)
group by pa2.STORE_NBR
```

Default to Metric Name

Default to Metric Name is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Default to Metric Name allows you to choose whether you want to use the metric name or a MicroStrategy-generated name as the column alias. When metric names are used, only the first 20 standard characters are used. If you have different metrics, the metric names start with the same 20 characters. It is hard to differentiate between the two, because they are always the same. The Default to Metric Name option does not work for some international customers.

If you choose to use the metric name and the metric name begins with a number, the letter M is attached to the beginning of the name during SQL generation. For example, a metric named 2003Revenue is renamed M2003Revenue. This occurs because Teradata does not allow a leading number in a metric name.

If you select the option **Use the metric name as the default metric column alias**, you should also set the maximum metric alias size. See *Default to Metric Name, page 105* below for information on setting this option.

Levels at which you can set this

Database instance only

Examples

Do not use the metric name as the default metric column alias (default)

```
insert into ZZTSU006VT7PO000
select a11.[MONTH_ID] AS MONTH_ID,
```

```
        a11.[ITEM_ID] AS ITEM_ID,
        a11.[EOH_QTY] AS WJXBFS1
from [INVENTORY_Q4_2003] a11,
        [LU_MONTH] a12,
        [LU_ITEM] a13
where a11.[MONTH_ID] = a12.[MONTH_ID] and
        a11.[ITEM_ID] = a13.[ITEM_ID]
 and            (a13.[SUBCAT_ID] in (25)
 and a12.[QUARTER_ID] in (20034))
```

Use the metric name as the default metric column alias

```
insert into ZZPO00
select a11.[MONTH_ID] AS MONTH_ID,
        a11.[ITEM_ID] AS ITEM_ID,
        a11.[EOH_QTY] AS Endonhand
from [{|Partition_Base_Table|}] a11,
        [LU_MONTH] a12,
        [LU_ITEM] a13
where a11.[MONTH_ID] = a12.[MONTH_ID] and
        a11.[ITEM_ID] = a13.[ITEM_ID]
 and            (a13.[SUBCAT_ID] in (25)
 and a12.[QUARTER_ID] in (20034))
```

Integer Constant in Metric

The Integer Constant in Metric property determines whether or not to add a ".0" after the integer. This prevents incorrect integer division, for example, 2/7 = 0. Normally a ".0" is added to an integer constant to have a float division (2.0/7.0 = 0.286). Some databases have trouble with this change, because some database functions only work with integer data types. This property allows you to turn OFF the addition of the ".0" if you have a database that does not properly handle the .0 added after the integer.

Levels at which you can set this

Database instance and metric

Join Across Datasets

Join Across Datasets is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Join Across Datasets determines how values for metrics are calculated when unrelated attributes from different datasets of a dashboard or document are included with metrics. For example, consider a dashboard with two separate datasets that include the following data:

The datasets are displayed below as simple grid visualizations within a dashboard.

Category	Region	Profit
Books	North	15
Electronics	North	30
Movies	South	20

Category	Sales
Books	70
Electronics	105
Movies	85

Notice that one dataset includes the Region attribute, however the other dataset only includes Category. The Region attribute is also not directly related to the Category attribute, but it is included with Category in one of the two datasets.

On this dashboard, you choose to create a new grid visualization with Region and Sales. These objects are not on the same dataset, so this requires combining the data from different datasets. By default, data is not joined for the unrelated attributes Category and Region, and the following data is displayed:

Region	Sales
North	260
South	260

The data for Sales is displayed as $260 for both Regions, which is the total sales of all regions. In most scenarios, this sales data should instead reflect the data for each region. This can be achieved by allowing data to be joined for the unrelated attributes Category and Region, which then displays the following data:

Now the data for Sales displays $185 for North (a combination of the sales for Books and Electronics, which were both for the North region) and $85 for South (sales for Movies, which was for the South region).

- **Disallow joins based on unrelated common attributes**: By default, data is not joined for unrelated attributes that are included on the same dataset. This option is to support backward compatibility.
- **Allow joins based on unrelated common attributes**: Data is joined for unrelated attributes that are included on the same dataset. This can allow metric data to consider unrelated attributes on the same dataset to logically combine the data, and thus provides results that are more accurate and intuitive in most cases.

Levels at which you can set this

Project and dashboard. To define this behavior for a dashboard open in Visual Insight, from the **File** menu, select **Document Properties**. You can then select to allow joins across datasets.

Max Metric Alias Size

Max Metric Alias Size is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Max Metric Alias Size allows you to set the maximum size of the metric alias string. This is useful for databases that only accept a limited number of characters for column names.

You should set the maximum metric alias size to fewer characters than your database's limit. This is because, in certain instances, such as when two column names are identical, the SQL engine adds one or more characters to one of the column names during processing to be able to differentiate between the names. Identical column names can develop when column names are truncated.

For example, if your database rejects any column name that is more than 30 characters and you set this VLDB property to limit the maximum metric alias size to 30 characters, the example presented by the following metric names still causes your database to reject the names during SQL processing:

- Sales Metric in Fairfax County for 2002
- Sales Metric in Fairfax County for 2003

The system limits the names to 30 characters based on the VLDB option you set in this example, which means that the metric aliases for both columns is as follows:

- SALESMETRICINFAIRFAXCOUNTYFOR2 (30 characters)
- SALESMETRICINFAIRFAXCOUNTYFOR21 (31 characters)

The SQL engine adds a 1 to one of the names because the truncated versions of both metric names are identical. That name is then 31 characters long and so the database rejects it.

Therefore, in this example you should use this feature to set the maximum metric alias size to fewer than 30 (perhaps 25), to allow room for the SQL engine to add one or two characters during processing in case the first 25 characters of any of your metric names are the same.

Levels at which you can set this

Database instance only

Metric Join Type

Metric Join Type is used to determine how to combine the result of one metric with that of other metrics. When this property is set to Outer Join, all the result rows of this metric are kept when combining results with other metrics. If there is only one metric on the report, this property is ignored.

There are multiple places to set this property:

- At the **DBMS** and **database instance** levels, it is set in the VLDB Properties Editor. This setting affects all the metrics in this project, unless it is overridden at a lower level.
- At the **metric** level, it can be set in either the VLDB Properties Editor or from the Metric Editor's **Tools** menu, and choosing **Metric Join Type**. The setting is applied in all the reports that include this metric.
- At the **report** level, it can be set from the Report Editor's **Data** menu, by pointing to **Report Data Options**, and choosing **Metric Join Type**. This setting overrides the setting at the metric level and is applied only for the currently selected report.

There is a related but separate property called Formula Join Type that can also be set at the metric level. This property is used to determine how to combine the result set together within this metric. This normally happens when a metric formula contains multiple facts that cause the Analytical Engine to use multiple fact tables. As a result,

sometimes it needs to calculate different components of one metric in different intermediate tables and then combine them. This property can only be set in the Metric Editor from the **Tools** menu, by pointing to **Advanced Settings**, and then choosing **Formula Join Type**.

Both Metric Join Type and Formula Join Type are used in the Analytical Engine to join multiple intermediate tables in the final pass. The actual logic is also affected by another VLDB property, Full Outer Join Support. When this property is set to YES, it means the corresponding database supports full outer join (92 syntax). In this case, the joining of multiple intermediate tables makes use of outer join syntax directly (left outer join, right outer join, or full outer join, depending on the setting on each metric/table). However, if the Full Outer Join Support is NO, then the left outer join is used to simulate a full outer join. This can be done with a union of the IDs of the multiple intermediate tables that need to do an outer join and then using the union table to left outer join to all intermediate tables, so this approach generates more passes. This approach was also used by MicroStrategy 6.x and earlier.

Also note that when the metric level is higher than the template level, the Metric Join Type property is normally ignored, unless you enable another property, Downward Outer Join Option. For detailed information, see *Relating column data with SQL: Joins, page 59*.

Levels at which you can set this

Database instance and metric

Non-Agg Metric Optimization

Non-Agg Metric Optimization is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Non-Agg Metric Optimization influences the behavior of non-aggregation metrics by either optimizing for smaller temporary tables or for less fact table access. This property can help improve query performance depending on the fact table size and the potential temporary table size. It may be more effective to create a larger temporary table so that you can avoid using the even larger fact table. If you are short on temporary table space or insert much data from the fact table into the temporary table, it may be better to use the fact table multiple times rather than create temporary tables. Your choice for this property depends on your data and report definitions.

Levels at which you can set this

Database instance, report, and template

Examples

Optimized for less fact table access (default)

The following example first creates a fairly large temporary table, but then never touches the fact table again.

```
select a11.REGION_NBR REGION_NBR,
```

```
a11.REGION_NBR REGION_NBR0,
a12.MONTH_ID MONTH_ID,
a11.DIVISION_NBR DIVISION_NBR,
a11.CUR_TRN_DT CUR_TRN_DT,
a11.TOT_SLS_DLR WJXBFS1
into ZZNB00
from REGION_DIVISION a11
join LOOKUP_DAY a12
on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
select pa1.REGION_NBR REGION_NBR,
pa1.MONTH_ID MONTH_ID,
min(pa1.CUR_TRN_DT) WJXBFS1
into ZZMB01
from ZZNB00 pa1
group by pa1.REGION_NBR,
pa1.MONTH_ID
select pa1.REGION_NBR REGION_NBR,
pa1.MONTH_ID MONTH_ID,
count(pa1.WJXBFS1) WJXBFS1
into ZZNC02
from ZZNB00 pa1
join ZZMB01 pa2
on (pa1.CUR_TRN_DT = pa2.WJXBFS1 and
pa1.MONTH_ID = pa2.MONTH_ID and
pa1.REGION_NBR = pa2.REGION_NBR)
group by pa1.REGION_NBR,
pa1.MONTH_ID
select distinct pa3.REGION_NBR REGION_NBR,
a13.REGION_DESC REGION_DESC,
a12.CUR_TRN_DT CUR_TRN_DT,
pa3.WJXBFS1 COUNTOFSALES
from ZZNC02 pa3
join LOOKUP_DAY a12
on (pa3.MONTH_ID = a12.MONTH_ID)
join LOOKUP_REGION a13
on (pa3.REGION_NBR = a13.REGION_NBR)
```

Optimized for smaller temp table

The following example does not create the large temporary table but must query the fact table twice.

```
select a11.REGION_NBR REGION_NBR,
a12.MONTH_ID MONTH_ID,
min(a11.CUR_TRN_DT) WJXBFS1
into ZZOP00
from REGION_DIVISION a11
join LOOKUP_DAY a12
on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
group by a11.REGION_NBR,
a12.MONTH_ID
select a11.REGION_NBR REGION_NBR,
a12.MONTH_ID MONTH_ID,
count(a11.TOT_SLS_DLR) COUNTOFSALES
into ZZMD01
from REGION_DIVISION a11
join LOOKUP_DAY a12
on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
join ZZOP00 pa1
on (a11.CUR_TRN_DT = pa1.WJXBFS1 and
a11.REGION_NBR = pa1.REGION_NBR and
a12.MONTH_ID = pa1.MONTH_ID)
group by a11.REGION_NBR,
a12.MONTH_ID
select distinct pa2.REGION_NBR REGION_NBR,
a13.REGION_DESC REGION_DESC,
```

```
a12.CUR_TRN_DT CUR_TRN_DT,
pa2.COUNTOFSALES COUNTOFSALES
from ZZMD01 pa2
join LOOKUP_DAY a12
on (pa2.MONTH_ID = a12.MONTH_ID)
join LOOKUP_REGION a13
on (pa2.REGION_NBR = a13.REGION_NBR)
```

NULL Check

NULL Check indicates how to handle arithmetic operations with NULL values. If NULL Check is enabled, the NULL2ZERO function is added, which changes NULL to 0 in any arithmetic calculation (+,-,*,/).

Levels at which you can set this

Database instance, report, and template

Separate COUNT DISTINCT

Separate Count Distinct is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Separate Count Distinct indicates how to handle COUNT (and other aggregation functions) when DISTINCT is present in the SQL.

Levels at which you can set this

Database instance, report, and template

Smart Metric Transformation

Smart Metric Transformation is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Due to the evaluation order used for smart metrics, compound metrics, and transformations, creating transformation metrics to display the variance or variance percentage of a smart metric or compound metric can return unexpected results in some scenarios.

For definitions and examples of smart metrics, compound metrics, and transformation metrics, see the Advanced Reporting Guide.

For example, the report sample shown below includes quarterly profit margins. Transformation metrics are included to display the last quarter's profit margin (`Last Quarter's (Profit Margin)`) and the variance of the profit margin and last quarter's profit margin (`((Profit Margin - (Last Quarter's (Profit Margin))`).

Quarter	Profit Margin	Last Quarter's (Profit Margin)	(Profit Margin) - (Last Quarter's (Profit Margin))
2010 Q2	14.82%	17.68%	(1.05%)
2010 Q3	15.33%	14.82%	18.46%
2010 Q4	13.42%	15.33%	0.78%
2011 Q1	17.64%	13.42%	(50.14%)
2011 Q2	14.98%	17.64%	(20.68%)
2011 Q3	15.07%	14.98%	15.68%
2011 Q4	13.31%	15.07%	(13.65%)
2012 Q1	17.64%	13.31%	(73.58%)
2012 Q2	14.63%	17.64%	(9.23%)
2012 Q3	15.33%	14.63%	26.21%
2012 Q4	13.65%	15.33%	5.62%

Since Profit Margin is a smart metric, the transformation metric that calculates the variance displays unexpected results. For example, consider the report row highlighted in the report example above. The profit margin for 2011 Q3 is 15.07% and the profit margin for 2011 Q2 is 14.98%. Both of these calculations are correct. However, an incorrect value of 15.68% is displayed as the variance.

You can modify the evaluation order to return correct variance results by defining the Smart Metric Transformation VLDB property as True. After making this change, the report displays the following results.

Quarter	Profit Margin	Last Quarter's (Profit Margin)	(Profit Margin) - (Last Quarter's (Profit Margin))
2010 Q2	14.82%	17.68%	(2.86%)
2010 Q3	15.33%	14.82%	0.52%
2010 Q4	13.42%	15.33%	(1.91%)
2011 Q1	17.64%	13.42%	4.22%
2011 Q2	14.98%	17.64%	(2.65%)
2011 Q3	15.07%	14.98%	0.09%
2011 Q4	13.31%	15.07%	(1.76%)
2012 Q1	17.64%	13.31%	4.33%
2012 Q2	14.63%	17.64%	(3.01%)
2012 Q3	15.33%	14.63%	0.70%
2012 Q4	13.65%	15.33%	(1.69%)

The variance is now displayed as 0.09%, which is the correct variance calculation (`15.07% - 14.98% = 0.09%`).

The Smart Metric Transformation VLDB property has the following options:

- **False** (default): Select this option for backwards compatibility with existing transformation metrics based on smart metrics or compound metrics.
- **True**: Select this option to modify the evaluation order to support transformation metrics that calculate a variance or variance percentage, based on the results of a smart metric or compound metric. Be aware that to apply this functionality to derived metrics you must select this option at the project level.

Levels at which you can set this

Project and metric

Subtotal Dimensionality Use

Subtotal Dimensionality Use is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Subtotal Dimensionality Use determines how the level of calculation is defined for metrics that are included on reports that use dynamic aggregation, which is an OLAP Services feature. This VLDB property has the following options:

- **Use only the grouping property of a level metric for dynamic aggregation** (default): The dimensionality, or level, of the metric is used to define how the metric data is calculated on the report when dynamic aggregation is also used. When selecting this option, only the grouping option for a level metric is used to calculate metric data. For detailed examples and information on defining the dimensionality of a metric, refer to the documentation on level metrics provided in the *Advanced Reporting Guide*.
- **Use only the grouping property of a level subtotal for dynamic aggregation**: The dimensionality, or level, of the metric's dynamic aggregation function is used to define how the metric data is calculated on the report when dynamic aggregation is also used. You can define the level of calculation for a metric's dynamic aggregation function by creating a subtotal, and then defining the level of calculation for that subtotal. When selecting this option, only the grouping option for a subtotal is used to calculate metric data. For detailed examples and information on creating subtotals, refer to the *Advanced Reporting Guide*.
- **Use both the grouping and filtering property of a level metric for dynamic aggregation**: The dimensionality, or level, of the metric is used to define how the metric data is calculated on the report when dynamic aggregation is also used. When selecting this option, both the grouping and filtering options for a level metric are used to calculate metric data. For detailed examples and information on defining the dimensionality of a metric, refer to the documentation on level metrics provided in the *Advanced Reporting Guide*.
- **Use both the grouping and filtering property of a level subtotal for dynamic aggregation**: The dimensionality, or level, of the metric's dynamic aggregation function is used to define how the metric data is calculated on the report when dynamic aggregation is also used. You can define the level of calculation for a metric's dynamic aggregation function by creating a subtotal, and then defining the level of calculation for that subtotal. When selecting this option, both the grouping and filtering options for a subtotal are used to calculate metric data. For detailed examples and information on creating subtotals, refer to the *Advanced Reporting Guide*.

Example

Consider a metric that performs a simple sum of cost data by using the following metric definition:

```
Sum(Cost) {~+}
```

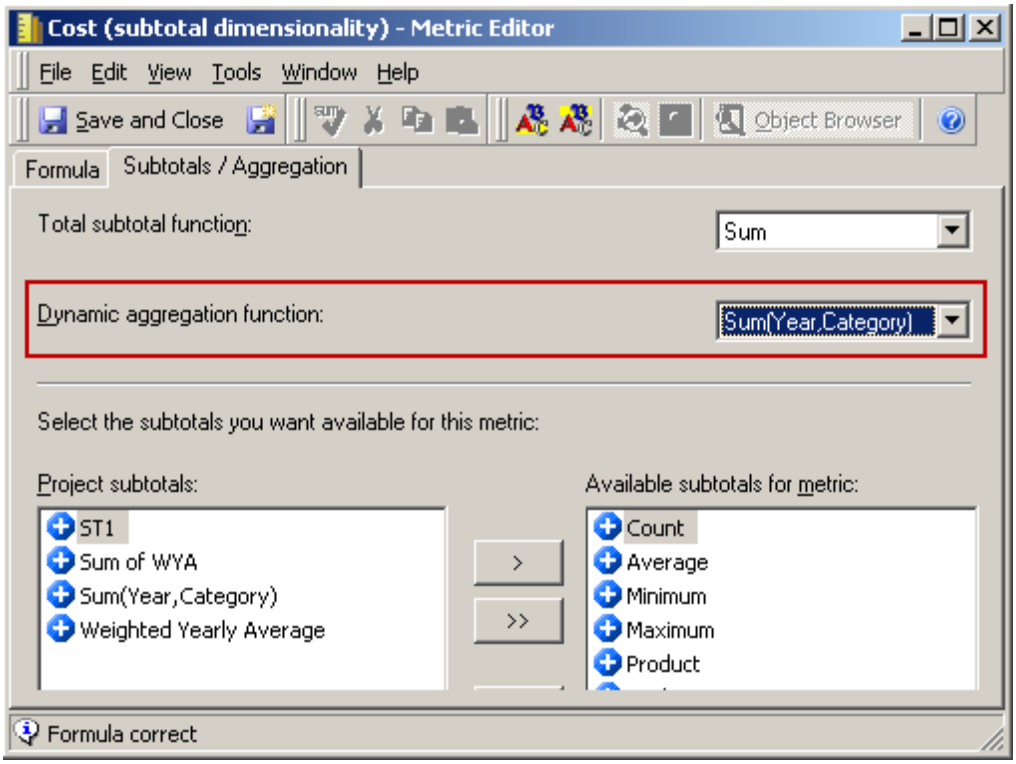
This metric is named Cost, and the syntax `{~+}` indicates that it calculates data at the level of the report it is included on. Another metric is created with the following metric definition:

```
Sum(Cost) {~+}
```

This metric also uses a subtotal for its dynamic aggregation function that uses the following definition:

```
Sum(x) {~+, !Year , !Category }
```

Notice that the function for this subtotal includes additional level information to perform the calculation based on the report level, Year, and Category. As shown in the image below, this subtotal function, named Sum(Year,Category) is applied as the metric's dynamic aggregation function.



This metric is named Cost (subtotal dimensionality). This metric along with the simple Cost metric is displayed on the report shown below, which also contains the attributes Year, Region, and Category.

Year	Region	Category	Metrics	Cost	Cost (subtotal dimensionality)
2009	Northeast	Books		$129,566	$129,566
		Electronics		$1,294,756	$1,294,756
		Movies		$246,473	$246,473
		Music		$235,538	$235,538
		Total		**$1,906,333**	**$1,906,333**
	Total			**$1,906,333**	**$1,906,333**
2010	Northeast	Books		$171,115	$171,115
		Electronics		$1,643,433	$1,643,433
		Movies		$314,602	$314,602
		Music		$305,441	$305,441
		Total		**$2,434,591**	**$2,434,591**
	Total			**$2,434,591**	**$2,434,591**
2011	Northeast	Books		$205,809	$205,809
		Electronics		$1,971,215	$1,971,215
		Movies		$377,415	$377,415
		Music		$358,320	$358,320
		Total		**$2,912,759**	**$2,912,759**
	Total			**$2,912,759**	**$2,912,759**
Total				**$7,253,683**	**$7,253,683**

Notice that the values for these two metrics are the same. This is because no dynamic aggregation is being performed, and the Subtotal Dimensionality Use VLDB property is also using the default option of Use dimensionality from metric for dynamic aggregation. With this default behavior still applied, the attribute Year can be removed from the grid of the report to trigger dynamic aggregation, as shown in the report below.

Region	Category	Metrics	Cost	Cost (subtotal dimensionality)
Northeast	Books		$506,490	$506,490
	Electronics		$4,909,404	$4,909,404
	Movies		$938,491	$938,491
	Music		$899,299	$899,299
	Total		**$7,253,683**	**$7,253,683**
Total			**$7,253,683**	**$7,253,683**

The metric values are still the same because both metrics are using the level of the metric. If the Subtotal Dimensionality Use VLDB property for the report is modified to use the option Use dimensionality from subtotal for dynamic aggregation, this affects the report results as shown in the report below.

Region	Category	Metrics	Cost	Cost (subtotal dimensionality)
Northeast	Books		$506,490	$7,253,683
	Electronics		$4,909,404	$7,253,683
	Movies		$938,491	$7,253,683
	Music		$899,299	$7,253,683
	Total		**$7,253,683**	**$7,253,683**
Total			**$7,253,683**	**$7,253,683**

The Cost (subtotal dimensionality) metric now applies the level defined in the subtotal function that is used as the metric's dynamic aggregation function. This displays the same Cost value for all categories in the Northeast region because the data is being returned as the total for all years and categories combined.

Levels at which you can set this

Database instance, report, template, and metric

Transformable AggMetric

The Transformable AggMetric VLDB property allows you to define what metrics should be used to perform transformations on compound metrics that use nested aggregation.

For example, you create two metrics. The first metric, referred to as Metric1, uses an expression of `Sum(Fact) {~+, Attribute+}`, where `Fact` is a fact in your project and `Attribute` is an attribute in your project used to define the level of Metric1. The second metric, referred to as Metric2, uses an expression of `Avg(Metric1){~+}`. Since both metrics use aggregation functions, Metric2 uses nested aggregation.

Including Metric2 on a report can return incorrect results for the following scenario:

- A transformation shortcut metric is defined on Metric2.
- Metric1 is defined at a lower level than the report level.

In this scenario, the transformation is applied to the outer metric, which in this case is Metric2. To perform the transformation correctly, the transformation should be applied for the inner metric, which in this case is Metric1. To apply the transformation to Metric1 in this scenario, you can use the Transformable AggMetric VLDB property. The options are:

- **False** (default): The metric uses default transformation behavior. This option should be used for all metrics except for those metrics that are defined for a scenario similar to Metric2 described above.
- **True**: The metric is defined as a metric to use to perform a transformation when it is included in another metric through nested aggregation. This option should be used only for metrics that are defined for a scenario similar to Metric2 described above.

Levels at which you can set this

Metric only

Transformation Role Processing

The Transformation Role Processing property is only available from the Transformation Editor. From the Transformation Editor, select **Schema Objects** and then choose **Transformations**. Right-click an object from the right pane and select **Edit**.

The Transformation Role Processing property lets you choose how transformation dates are calculated when there are multiple attributes to transform. The example below

considers the common Day, Week, and Month schema setup. The schema has Week and Month as a parent to Day. Week and Month are unrelated. This Month, Week, and Day hierarchy setup is a common scenario where this property makes a difference.

Example

You have a report with Week, Sales, and Last Year Sales on the template, filtered by Month. The default behavior is to calculate the Last Year Sales with the following SQL. Notice that the date transformation is done for Month and Week.

```
insert into ZZT6T02D01
select a14.DAT_YYYYWW DAT_YYYYWW,
 sum(a11.SALES) SALESLY
from FT1 a11
 join TRANS_DAY a12
   on (a11.DAT_YYYYMMDD = a12.DAT_YYYYMMDD)
 join TRANS_DAY_MON a13
   on (a12.DAT_YYYYYYMM = a13.DAT_LYM)
 join TRANS_DAY_WEEK a14
   on (a12.DAT_YYYYWW = a14.DAT_LYW)
where a13.DAT_YYYYMM in (200311)
group by a14.DAT_YYYYWW
```

The new behavior applies transformation only to the highest common child when it is applicable to multiple attributes. The SQL is shown in the following syntax. Notice that the date transformation is done only at the Day level, because Day is the highest common child of Week and Month. So the days are transformed, and then you filter for the correct Month, and then Group by Week.

```
insert into ZZT6T02D01
select a12.DAT_YYYYWW DAT_YYYYWW,
 sum(a11.SALES) SALESLY
from FT1 a11
 join TRANS_DAY a12
   on (a11.DAT_YYYYMMDD = a12.DAT_YYYYMMLYT)
where a12.DAT_YYYYMM in (200311)
group by a12.DAT_YYYYWW
```

Zero Check

Zero Check indicates how to handle division by zero. If zero checking is enabled, the ZERO2NULL function is added, which changes 0 to NULL in the denominator of any division calculation.

Levels at which you can set this

Database instance, report, and template

Customizing SQL statements: Pre/Post Statements

The table below summarizes the Pre/Post Statements VLDB properties. Additional details about each property, including examples and a list of wild cards, are available by clicking on the links in the table.

Property	Description	Possible Values	Default Value
Cleanup Post Statement	Appends string after final drop statement.	User-defined	NULL
Data mart SQL to be executed after data mart creation	SQL statements included after the CREATE statement used to create the data mart.	User-defined	NULL
Data mart SQL to be executed before inserting data	SQL statements included before the INSERT statement used to insert data into the data mart.	User-defined	NULL
Data mart SQL to be executed prior to data mart creation	SQL statements included before the CREATE statement used to create the data mart.	User-defined	NULL
Drop Database Connection	Defines whether the database connection is dropped after user-defined SQL is executed on the database.	• Drop database connection after running user-defined SQL • Do not drop database connection after running user-defined SQL	Drop database connection after running user-defined SQL
Element Browsing Post Statement	SQL statements issued after element browsing requests.	User-defined	NULL
Element Browsing Pre Statement	SQL statements issued before element browsing requests.	User-defined	NULL
Insert Mid Statement 1-5	SQL statements issued between multiple insert statements. For the first four statements, each contains single SQL. The last statement can contain multiple SQL statements concatenated by ";".	User-defined	NULL
Insert Post Statement 1-5	SQL statements issued after create, after first insert only for explicit temp table creation. For the first four statements, each contains single SQL. The last statement can contain multiple SQL statements concatenated by ";".	User-defined	NULL

Property	Description	Possible Values	Default Value
Insert Pre Statement 1-5	SQL statements issued after create before first insert only for explicit temp table creation. For the first four statements, each contains single SQL. The last statement can contain multiple SQL statements concatenated by ";".	User-defined	NULL
Report Post Statement 1-5	SQL statements issued after report requests. For the first four statements, each contains single SQL. The last statement can contain multiple SQL statements concatenated by ";".	User-defined	NULL
Report Pre Statement 1-5	SQL statements issued before report requests. For the first four statements, each contains single SQL. The last statement can contain multiple SQL statements concatenated by ";".	User-defined	NULL
Table Post Statement 1-5	SQL statements issued after creating new table and insert. For the first four statements, each contains single SQL. The last statement can contain multiple SQL statements concatenated by ";".	User-defined	NULL
Table Pre Statement 1-5	SQL statements issued before creating new table. For the first four statements, each contains single SQL. The last statement can contain multiple SQL statements concatenated by ";".	User-defined	NULL

You can insert the following syntax into strings to populate dynamic information by the SQL Engine:

- **!!!** inserts column names, separated by commas (can be used in Table Pre/Post and Insert Pre/Mid statements).
- **!!** inserts an exclamation (!) (can be used in Table Pre/Post and Insert Pre/Mid statements). Note that "!!=" inserts a not equal to sign in the SQL statement.
- **???** inserts the table name (can be used in Data Mart Insert/Pre/Post statements, Insert Pre/Post, and Table Post statements).
- **;;** inserts a semicolon (;) in Statement5 (can be used in all Pre/Post statements). Note that a single ";" (semicolon) acts as a separator.
- **!a** inserts column names for attributes only (can be used in Table Pre/Post and Insert Pre/Mid statements).
- **!d** inserts the date (can be used in all Pre/Post statements).
- **!f** inserts the report path (can be used in all Pre/Post statements except Element Browsing). An example is: `\MicroStrategy Tutorial\Public Objects\Reports\MicroStrategy Platform Capabilities\Ad hoc Reporting\Sorting\Yearly Sales`

- **!i** inserts the job priority of the report which is represented as an integer from 0 to 999 (can be used in all Pre/Post statements).
- **!o** inserts the report name (can be used in all Pre/Post statements).
- **!u** inserts the user name (can be used in all Pre/Post statements).
- **!j** inserts the Intelligence Server Job ID associated with the report execution (can be used in all Pre/Post statements).
- **!r** inserts the report GUID, the unique identifier for the report object that is also available in the Enterprise Manager application (can be used in all Pre/Post statements).
- **!t** inserts a timestamp (can be used in all Pre/Post statements).
- **!p** inserts the project name with spaces omitted (can be used in all Pre/Post statements).
- **!z** inserts the project GUID, the unique identifier for the project (can be used in all Pre/Post statements).
- **!s** inserts the user session GUID, the unique identifier for the user's session that is also available in the Enterprise Manager application (can be used in all Pre/Post statements).
- The # character is a special token that is used in various patterns and is treated differently than other characters. One single # is absorbed and two # are reduced to a single #. For example to show three # characters in a statement, enter six # characters in the code. You can get any desired string with the right number of # characters. Using the # character is the same as using the **;** character.

The table below shows the location of some of the most important VLDB/DSS settings in a Structured Query Language (SQL) query structure. If the properties in the table are set, the values replace the corresponding tag in the query:

Tag	VLDB properties (MSTR 7.x)
<1>	Report PreStatement (1-5)
<2>	Table PreStatement (1-5)
<3>	Table Qualifier
<4>	Table Descriptor
<5>	Table Prefix
<6>	Table Option
<7>	Table Space
<8>	Create PostString
<9>	Pre DDL COMMIT

Tag	VLDB properties (MSTR 7.x)
<10>	Insert PreStatement (1-5)
<11>	Insert Table Option
<12>	SQL Hint
<13>	Post DDL COMMIT
<14>	Insert PostString
<15>	Insert MidStatement (1-5)
<16>	Table PostStatement (1-5)
<17>	Index Qualifier
<18>	Index PostString
<19>	Select PostString
<20>	Report PostStatement (1-5)
<21>	Commit after Final Drop
<22>	Cleanup PostStatement

Query structure

```
<1>
     <2>
          CREATE <3> TABLE <4> <5><table name> <6>
          (<fields' definition>)
          <7>
          <8>
          <9>(COMMIT)
     <10>
          INSERT INTO <5><table name><11>
          SELECT <12> <fields list>
          FROM <tables list>
          WHERE <joins and filter>
          <13>(COMMIT)
     <14>
     <15>
     <16>
          CREATE <17> INDEX <index name> ON
                                <fields list>
     <18>
          SELECT <12> <fields list>
          FROM <tables list>
          WHERE <joins and filter>
     <19>
<20>
          DROP TABLE TABLENAME
<21>
<22>
```

The Commit after Final Drop property (<21>) is sent to the warehouse even if the SQL View for the report does not show it.

Cleanup Post Statement

The Cleanup Post Statement property allows you to insert your own SQL string after the final DROP statement. There are five settings, numbered 1-5. Each text string entered in Cleanup Post Statement 1 through Cleanup Post Statement 4 is executed separately as a single statement. To execute more than 5 statements, insert multiple statements in Cleanup Post Statement 5, separating each statement with a ";". The SQL Engine then breaks it into individual statements using ";" as the separator and executes the statements separately.

Levels at which you can set this

Database instance, report, and template

Example

In the following example the setting values are:

```
Cleanup Post Statement1=/* Cleanup Post Statement1 */
Create table TABLENAME
(ATTRIBUTE_COL1 VARCHAR(20),
  FORM_COL2 CHAR(20),
  FACT_COL3 FLOAT)
 primary index (ATTRIBUTE_COL1, FORM_COL2)
insert into TABLENAME
select A1.COL1,
         A2.COL2,
         A3.COL3
from TABLE1 A1,
        TABLE2 A2,
        TABLE3 A3
where A1.COL1 = A2.COL1 and A2.COL4=A3.COL5
insert into TABLENAME
select A1.COL1,
         A2.COL2,
         A3.COL3
from TABLE4 A1,
        TABLE5 A2,
        TABLE6 A3
where A1.COL1 = A2.COL1 and A2.COL4=A3.COL5

create index IDX_TEMP1(STORE_ID, STORE_DESC)
select A1.STORE_NBR,
max(A1.STORE_DESC)
from LOOKUP_STORE
Where A1 A1.STORE_NBR = 1
group by A1.STORE_NBR
drop table TABLENAME
/* Cleanup Post Statement 1*/
```

Data mart SQL to be executed after data mart creation

The Data mart SQL to be executed after data mart creation VLDB property allows you to define SQL statements that are included after data mart creation. These SQL statements are included after the `CREATE` statement for the data mart table. This allows you to customize the statement used to create data marts.

Levels at which you can set this

Database instance and data mart

Data mart SQL to be executed before inserting data

The Data mart SQL to be executed before inserting data VLDB property allows you to define SQL statements issued before inserting data into a data mart. These SQL statements are included before the `INSERT` statement for the data mart table. This allows you to customize the statement used to insert data into data marts.

Levels at which you can set this

Database instance and data mart

Data mart SQL to be executed prior to data mart creation

The Data mart SQL to be executed prior to data mart creation VLDB property allows you to define SQL statements that are included before data mart creation. These SQL statements are included before the `CREATE` statement for the data mart table. This allows you to customize the statement used to create data marts.

Levels at which you can set this

Database instance and data mart

Drop Database Connection

The Drop Database Connection VLDB property allows you to define whether the database connection is dropped after user-defined SQL is executed on the database. This VLDB property has the following options:

- **Drop database connection after running user-defined SQL** (default): The database connection is dropped after user-defined SQL is executed on the database. This ensures that database connections are not left open and unused for extended periods of time after user-defined SQL is executed.
- **Do not drop database connection after running user-defined SQL**: The database connection remains open after user-defined SQL is executed on the database. This can keep the database connection open for additional user-defined SQL statements to be executed.

Levels at which you can set this

Database instance, report, and template

Element Browsing Post Statement

The Element Browsing Post Statement VLDB property is used to insert custom SQL statements after the completion of all element browsing requests. For example, an element browsing request occurs when a user expands an attribute to view its attribute elements.

Including SQL statements after the completion of element browsing requests can allow you to define the priority of element browsing requests to be higher or lower than the priority for report requests. You can also include any other SQL statements required to better support element browsing requests. You can include multiple statements to be executed. Each statement must be separated by a semicolon (`;`). The SQL Engine then executes the statements separately.

If you modify the Element Browsing PostStatement VLDB property, the statements defined in the Report Post Statement VLDB property are not used for element browsing requests. Priority of report requests and other post-report SQL statements can be defined using the Report Post Statement VLDB properties, which are described in *Element Browsing Post Statement, page 124*.

Levels at which you can set this

Database instance only

Element Browsing Pre Statement

The Element Browsing Pre Statement VLDB property is used to insert custom SQL statements at the beginning of all element browsing requests. For example, an element browsing request occurs when a user expands an attribute to view its attribute elements.

Including SQL statements prior to element browsing requests can allow you to define the priority of element browsing requests to be higher or lower than the priority for report requests. You can also include any other SQL statements required to better support element browsing requests. You can include multiple statements to be executed, separated by a semicolon (`;`). The SQL Engine then executes the statements separately.

If you do modify the Element Browsing PreStatement VLDB property, the statements defined in the Report Pre Statement VLDB property are not used for element browsing requests. Priority of report requests and other pre-report SQL statements can also be defined using the Report Pre Statement VLDB properties, which are described in *Element Browsing Pre Statement, page 124*.

Levels at which you can set this

Database instance only

Insert Mid Statement

The Insert Mid Statement property is used to insert your own custom SQL strings between the first INSERT INTO SELECT statement and subsequent INSERT INTO SELECT statements inserting data into the same table. There are five settings in total,

numbered 1-5. Each text string entered in Insert Mid Statement 1 through Insert Mid Statement 4 is executed separately as a single statement. To execute more than 5 statements, you can put multiple statements in Insert Mid Statement 5, separating each statement with a ";". The SQL Engine then breaks it into individual statements using ";" as the separator and executes the statements separately.

Multiple INSERT INTO SELECT statements to the same table occur in reports involving partition tables and outer joins. The UNION Multiple Inserts VLDB property affects this property. If the UNION Multiple Inserts VLDB property is set to Use Union, there is only one insert into the intermediate table. This setting is applicable when the Intermediate Table Type VLDB property is set to Permanent or Temporary table.

Levels at which you can set this

Database instance, report, and template

Examples

In the following example, the setting values are:

```
Insert MidStatement1=/* ??? Insert MidStatement1 */
```

UNION Multiple Inserts = Do Not Use UNION

```
select a11.PBTNAME PBTNAME
from HARI_STORE_ITEM_PTMAP a11
create table ZZTIS00H5YAPO000 (
                ITEM_NBR DECIMAL(10, 0),
                CLASS_NBR DECIMAL(10, 0),
                STORE_NBR DECIMAL(10, 0),
                XKYCGT INTEGER,
                TOTALSALES FLOAT)
insert into ZZTIS00H5YAPO000
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                0 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from            HARI_STORE_ITEM_93 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
/* ZZTIS00H5YAPO000 Insert MidStatement1 */
insert into ZZTIS00H5YAPO000
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                1 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_ITEM_94 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
select pa1.ITEM_NBR ITEM_NBR,
                pa1.CLASS_NBR CLASS_NBR,
                max(a11.ITEM_DESC) ITEM_DESC,
                max(a11.CLASS_DESC) CLASS_DESC,
                pa1.STORE_NBR STORE_NBR,
                max(a12.STORE_DESC) STORE_DESC,
                sum(pa1.TOTALSALES) TOTALSALES
```

```
from ZTIS00H5YAPO000 pa1
                join HARI_LOOKUP_ITEM a11
                  on (pa1.CLASS_NBR = a11.CLASS_NBR and
                pa1.ITEM_NBR = a11.ITEM_NBR)
                join HARI_LOOKUP_STORE a12
                  on (pa1.STORE_NBR = a12.STORE_NBR)
group by pa1.ITEM_NBR,
                pa1.CLASS_NBR,
                pa1.STORE_NBR
```

UNION Multiple Inserts = Use UNION

```
select           a11.PBTNAME PBTNAME
from            HARI_STORE_ITEM_PTMAP a11
create table ZZTIS00H5YEPO000 (
                ITEM_NBR DECIMAL(10, 0),
                CLASS_NBR DECIMAL(10, 0),
                STORE_NBR DECIMAL(10, 0),
                XKYCGT INTEGER,
                TOTALSALES FLOAT)
insert into ZZTIS00H5YEPO000
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                0 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_ITEM_93 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
union all
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                1 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_ITEM_94 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
select pa1.ITEM_NBR ITEM_NBR,
                pa1.CLASS_NBR CLASS_NBR,
                max(a11.ITEM_DESC) ITEM_DESC,
                max(a11.CLASS_DESC) CLASS_DESC,
                pa1.STORE_NBR STORE_NBR,
                max(a12.STORE_DESC) STORE_DESC,
                sum(pa1.TOTALSALES) TOTALSALES
from ZZTIS00H5YEPO000 pa1
                join HARI_LOOKUP_ITEM a11
                  on (pa1.CLASS_NBR = a11.CLASS_NBR and
                pa1.ITEM_NBR = a11.ITEM_NBR)
                join HARI_LOOKUP_STORE a12
                  on (pa1.STORE_NBR = a12.STORE_NBR)
group by pa1.ITEM_NBR,
                pa1.CLASS_NBR,
                pa1.STORE_NBR
```

Insert Post Statement

This property is used to insert your custom SQL statements after CREATE and after the first INSERT INTO SELECT statement for explicit temp table creation. There are five settings, numbered 1-5. Each text string entered in Insert Post Statement 1 through

Insert Post Statement 4 is executed separately as a single statement. To execute more than 5 statements, insert multiple statement in Insert Post Statement 5, separating each statement with a ";". The SQL Engine then breaks it into individual statements using ";" as the separator and executes the statements separately.

Multiple INSERT INTO SELECT statements to the same table occur in reports involving partition tables and outer joins. The UNION Multiple Inserts VLDB property does not affect this property, but the Table Creation Type property does. The Table Creation Type property is applicable when the Intermediate Table Type VLDB property is set to Permanent or Temporary table.

Levels at which you can set this

Database instance, report, and template

Example

In the following example, the setting values are:

```
Insert PostStatement1=/* ??? Insert PostStatement1 */
```

Table Creation Type= Explicit

```
select a11.PBTNAME PBTNAME
from            HARI_STORE_ITEM_PTMAP a11
create table ZZTIS00H601PO000 (
                ITEM_NBR DECIMAL(10, 0),
                CLASS_NBR DECIMAL(10, 0),
                STORE_NBR DECIMAL(10, 0),
                XKYCGT INTEGER,
                TOTALSALES FLOAT)
insert into ZZTIS00H601PO000
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                0 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_ITEM_93 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
insert into ZZTIS00H601PO000
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                1 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_ITEM_94 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
/* ZZTIS00H601PO000 Insert PostStatement1 */
select pa1.ITEM_NBR ITEM_NBR,
                pa1.CLASS_NBR CLASS_NBR,
                max(a11.ITEM_DESC) ITEM_DESC,
                max(a11.CLASS_DESC) CLASS_DESC,
                pa1.STORE_NBR STORE_NBR,
                max(a12.STORE_DESC) STORE_DESC,
                sum(pa1.TOTALSALES) TOTALSALES
from ZZTIS00H601PO000 pa1
                join HARI_LOOKUP_ITEM a11
```

```
                  on (pa1.CLASS_NBR = a11.CLASS_NBR and
                pa1.ITEM_NBR = a11.ITEM_NBR)
                join HARI_LOOKUP_STORE a12
                 on (pa1.STORE_NBR = a12.STORE_NBR)
group by pa1.ITEM_NBR,
                pa1.CLASS_NBR,
                pa1.STORE_NBR
```

Table Creation Type= Implicit

```
select a11.PBTNAME PBTNAME
from HARI_STORE_ITEM_PTMAP a11
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                0 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
into ZZTIS00H60BPO000
from HARI_STORE_ITEM_93 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
insert into ZZTIS00H60BPO000
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                1 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_ITEM_94 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
select pa1.ITEM_NBR ITEM_NBR,
                pa1.CLASS_NBR CLASS_NBR,
                max(a11.ITEM_DESC) ITEM_DESC,
                max(a11.CLASS_DESC) CLASS_DESC,
                pa1.STORE_NBR STORE_NBR,
                max(a12.STORE_DESC) STORE_DESC,
                sum(pa1.TOTALSALES) TOTALSALES
from ZTIS00H60BPO000 pa1
                join HARI_LOOKUP_ITEM a11
                  on (pa1.CLASS_NBR = a11.CLASS_NBR and
                pa1.ITEM_NBR = a11.ITEM_NBR)
                join HARI_LOOKUP_STORE a12
                  on (pa1.STORE_NBR = a12.STORE_NBR)
group by pa1.ITEM_NBR,
                pa1.CLASS_NBR,
                pa1.STORE_NBR
```

Insert Pre Statement

The Insert Pre Statement property is used to insert your custom SQL statements after CREATE but before the first INSERT INTO SELECT statement for explicit temp table creation. There are five settings, numbered 1-5. Each text string entered in Insert Pre Statement 1 through Insert Pre Statement 4 is executed separately as a single statement. To execute more than 5 statements, insert multiple statements in Insert Pre Statement 5, separating each statement with a ";". The SQL Engine then breaks it into individual statements using ";" as the separator and executes the statements separately.

Multiple INSERT INTO SELECT statements to the same table occur in reports involving partition tables and outer joins. The UNION Multiple Inserts VLDB property does not

affect this property, but the Table Creation Type property does. The Table Creation Type property is applicable when the Intermediate Table Type VLDB property is set to Permanent or Temporary table.

Levels at which you can set this

Database instance, report, and template

Examples

In the following examples, the setting values are:

```
Insert PreStatement1=/* ??? Insert
PreStatement1 */
```

Table Creation Type= Explicit

```
select a11.PBTNAME PBTNAME
from HARI_STORE_ITEM_PTMAP a11
create table ZZTIS00H601PO000 (
                ITEM_NBR DECIMAL(10, 0),
                CLASS_NBR DECIMAL(10, 0),
                STORE_NBR DECIMAL(10, 0),
                XKYCGT INTEGER,
                TOTALSALES FLOAT)
/* ZZTIS00H601PO000 Insert PreStatement1 */
insert into ZZTIS00H601PO000
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                0 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_ITEM_93 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
insert into ZZTIS00H601PO000
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                1 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_ITEM_94 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
select pa1.ITEM_NBR ITEM_NBR,
                pa1.CLASS_NBR CLASS_NBR,
                max(a11.ITEM_DESC) ITEM_DESC,
                max(a11.CLASS_DESC) CLASS_DESC,
                pa1.STORE_NBR STORE_NBR,
                max(a12.STORE_DESC) STORE_DESC,
                sum(pa1.TOTALSALES) TOTALSALES
from ZZTIS00H601PO000 pa1
                join HARI_LOOKUP_ITEM a11
                 on (pa1.CLASS_NBR = a11.CLASS_NBR and
                pa1.ITEM_NBR = a11.ITEM_NBR)
                join HARI_LOOKUP_STORE a12
                  on (pa1.STORE_NBR = a12.STORE_NBR)
group by pa1.ITEM_NBR,
                pa1.CLASS_NBR,
                pa1.STORE_NBR
```

Table Creation Type= Implicit

```
select a11.PBTNAME PBTNAME
from HARI_STORE_ITEM_PTMAP a11
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                0 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
into ZZTIS00H60BPO000
from HARI_STORE_ITEM_93 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
insert into ZZTIS00H60BPO000
select a11.ITEM_NBR ITEM_NBR,
                a11.CLASS_NBR CLASS_NBR,
                a11.STORE_NBR STORE_NBR,
                1 XKYCGT,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_ITEM_94 a11
group by a11.ITEM_NBR,
                a11.CLASS_NBR,
                a11.STORE_NBR
select pa1.ITEM_NBR ITEM_NBR,
                pa1.CLASS_NBR CLASS_NBR,
                max(a11.ITEM_DESC) ITEM_DESC,
                max(a11.CLASS_DESC) CLASS_DESC,
                pa1.STORE_NBR STORE_NBR,
                max(a12.STORE_DESC) STORE_DESC,
                sum(pa1.TOTALSALES) TOTALSALES
from ZZTIS00H60BPO000 pa1
                join HARI_LOOKUP_ITEM a11
                  on (pa1.CLASS_NBR = a11.CLASS_NBR and
                pa1.ITEM_NBR = a11.ITEM_NBR)
                join HARI_LOOKUP_STORE a12
                  on (pa1.STORE_NBR = a12.STORE_NBR)
group by pa1.ITEM_NBR,
                pa1.CLASS_NBR,
```

Report Post Statement

The Report Post Statement property is used to insert custom SQL statements after the final SELECT statement but before the DROP statements. There are five settings, numbered 1-5. Each text string entered in Report Post Statement 1 through Report Post Statement 4 is executed separately as a single statement. To execute more than 5 statements, insert multiple statements in Report Post Statement 5, separating each statement with a ";". The SQL Engine then breaks them into individual statements using ";" as the separator and executes the statements separately.

If you do not modify the Element Browsing Post Statement VLDB property, the statements defined in this Report Post Statement VLDB property are also used for element browsing requests. For example, an element browsing request occurs when a user expands an attribute to view its attribute elements. To define statements that apply only to element browsing requests, see *Report Post Statement, page 130*.

Levels at which you can set this

Database instance, report, and template

Example

In the following example, the setting values are:

```
Create table TABLENAME
(ATTRIBUTE_COL1 VARCHAR(20),
  FORM_COL2 CHAR(20),
  FACT_COL3 FLOAT)
 primary index (ATTRIBUTE_COL1, FORM_COL2)
insert into TABLENAME
select A1.COL1,
         A2.COL2,
         A3.COL3
from TABLE1 A1,
        TABLE2 A2,
        TABLE3 A3
where A1.COL1 = A2.COL1 and A2.COL4=A3.COL5
insert into TABLENAME
select A1.COL1,
         A2.COL2,
         A3.COL3
from TABLE4 A1,
        TABLE5 A2,
        TABLE6 A3
where A1.COL1 = A2.COL1 and A2.COL4=A3.COL5

create index IDX_TEMP1(STORE_ID, STORE_DESC)
select A1.STORE_NBR,
max(A1.STORE_DESC)
from LOOKUP_STORE
Where A1 A1.STORE_NBR = 1
group by A1.STORE_NBR
/* Report Post Statement 1*/
drop table TABLENAME
```

Report Pre Statement

The Report Pre Statement property is used to insert custom SQL statements at the beginning of the Report SQL. There are five settings, numbered 1-5. Each text string entered in Report Pre Statement 1 through Report Pre Statement 4 is executed separately as a single statement. To execute more than 5 statements, insert multiple statements in Report Pre Statement 5, separating each statement with a ";". The SQL Engine then breaks them into individual statements using ";" as the separator and executes the statements separately.

If you do not modify the Element Browsing Pre Statement VLDB property, the statements defined in this Report Pre Statement VLDB property are also used for element browsing requests. For example, an element browsing request occurs when a user expands an attribute to view its attribute elements. To define statements that apply only to element browsing requests, see *Report Pre Statement, page 131*.

Levels at which you can set this

Database instance, report, and template

Example

In the following example, the setting values are:

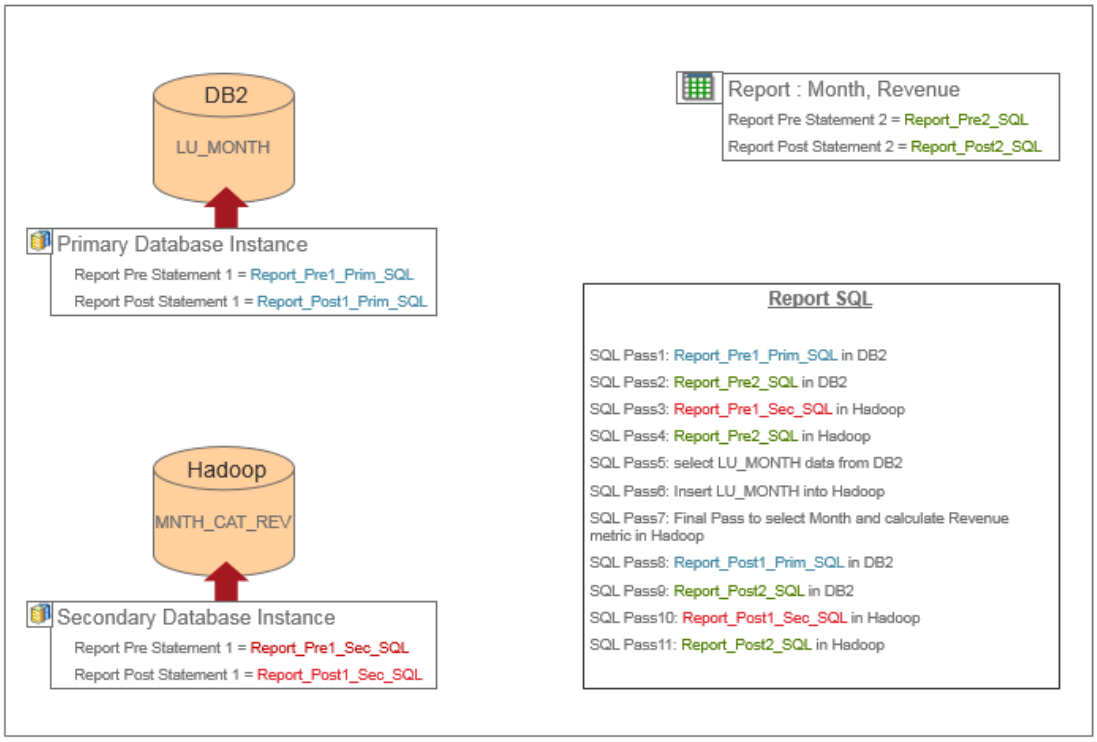
```
Report Pre Statement1=/* Report Pre Statement1 */
/* Report Pre Statement 1*/
Create table TABLENAME
(ATTRIBUTE_COL1 VARCHAR(20),
  FORM_COL2 CHAR(20),
  FACT_COL3 FLOAT)
 primary index (ATTRIBUTE_COL1, FORM_COL2)
insert into TABLENAME
select A1.COL1,
         A2.COL2,
         A3.COL3
from TABLE1 A1,
        TABLE2 A2,
        TABLE3 A3
where A1.COL1 = A2.COL1 and A2.COL4=A3.COL5
insert into TABLENAME
select A1.COL1,
         A2.COL2,
         A3.COL3
from TABLE4 A1,
        TABLE5 A2,
        TABLE6 A3
where A1.COL1 = A2.COL1 and A2.COL4=A3.COL5

create index IDX_TEMP1(STORE_ID, STORE_DESC)
select A1.STORE_NBR,
max(A1.STORE_DESC)
from LOOKUP_STORE
Where A1 A1.STORE_NBR = 1
group by A1.STORE_NBR
drop table TABLENAME
```

Multi-source Report Pre and Post Statements

Report Pre and Post SQL statements can be applied and executed against multiple databases used as a data source for reports. The Pre and Post SQL statements do not have to be placed in a particular order in the overall query as they operate independently from one another.

Report pre and post statements will not execute when a report or dashboard is built when attributes are added using Browse All Objects.

The example below shows an instance of how pre and post statements at both the report level and database instance level are applied and executed against multiple sources.



For examples of the syntax required for these statements, see the *Report Pre Statement* and *Report Post Statement* sections.

Table Post Statement

The Table Post Statement property is used to insert custom SQL statements after the CREATE TABLE and INSERT INTO statements. There are five settings, numbered 1-5. Each text string entered in Table Post Statement 1 through Table Post Statement 4 is executed separately as a single statement. To execute more than 5 statements, insert multiple statements in Table Post Statement 5, separating each statement with a ";". The SQL Engine then breaks them into individual statements using ";" as the separator and executes the statements separately. This property is applicable when the Intermediate Table Type VLDB property is set to Permanent or Temporary table or Views. The custom SQL is applied to every intermediate table or view.

Levels at which you can set this

Database instance, report, and template

Example

In the following example, the setting values are:

```
Table PostStatement1=/* ??? Table PostStatement1 */
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
into #ZZTIS00H63PMQ000
from HARI_STORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
```

```
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
/* #ZZTIS00H63PMQ000 Table PostStatement 1*/
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
        max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
        a11.STORE_NBR STORE_NBR,
        max(a13.STORE_DESC) STORE_DESC,
        sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_DEPARTMENT a11
        join #ZZTIS00H63PMQ000 pa1
         on (a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
         a11.STORE_NBR = pa1.STORE_NBR)
        join HARI_LOOKUP_DEPARTMENT a12
         on (a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR)
        join HARI_LOOKUP_STORE a13
         on (a11.STORE_NBR = a13.STORE_NBR)
group by a11.DEPARTMENT_NBR,
        a11.STORE_NBR
```

Table Pre Statement

The Table Pre Statement property is used to insert custom SQL statements before the CREATE TABLE statement. There are five settings, numbered 1-5. Each text string entered in Table Pre Statement 1 through Table Pre Statement 4 is executed separately as a single statement. To execute more than 5 statements, insert multiple statements in Table Pre Statement 5, separating each statement with a ";". The SQL Engine then breaks them into individual statements using ";" as the separator and executes the statements separately. This property is applicable when the Intermediate Table Type VLDB property is set to Permanent or Temporary table or Views. The custom SQL is applied to every intermediate table or view.

Levels at which you can set this

Database instance, report, and template

Example

In the following example, the setting values are:

```
Table PreStatement1=/* ??? Table
PreStatement1 */
/*Table PreStatement 1*/
create table ZZTIS00H63RMQ000 (
                DEPARTMENT_NBR          DECIMAL(10, 0),
                STORE_NBR               DECIMAL(10, 0))
insert into ZZTIS00H63RMQ000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
from HARI_STORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
        max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
        a11.STORE_NBR STORE_NBR,
        max(a13.STORE_DESC) STORE_DESC,
        sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_DEPARTMENT a11
        join ZZTIS00H63RMQ000 pa1
```

```
        on (a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
        a11.STORE_NBR = pa1.STORE_NBR)
        join HARI_LOOKUP_DEPARTMENT a12
        on  (a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR)
        join HARI_LOOKUP_STORE a13
        on (a11.STORE_NBR = a13.STORE_NBR)
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Optimizing queries

The table below summarizes the Query Optimizations VLDB properties. Additional details about each property, including examples where necessary, are provided in the sections following the table.

Property	Description	Possible Values	Default Value
Additional Final Pass Option	Determines whether the Engine calculates an aggregation function and a join in a single pass or in separate passes in the SQL.	• (default) Final pass CAN do aggregation and join lookup tables in one pass • One additional final pass only to join lookup tables	Final pass CAN do aggregation and join lookup tables in one pass
Apply Filter Options	Indicates during which pass the report filter is applied.	• Apply filter only to passes touching warehouse tables • Apply filter to passes touching warehouse tables and last join pass, if it does a downward join from the temp table level to the template level • Apply filter to passes touching warehouse tables and last join pass	Apply filter only to passes touching warehouse tables
Attribute Element Number Count Method	Controls how the total number of rows are calculated for incremental fetch.	• Use Count(Attribute@ID) to calculate total element number (uses count distinct if necessary) • Use ODBC cursor to calculate total element number	Use Count (Attribute@ID) to calculate total element number (uses count distinct if necessary) For Tandem databases, the default is Use ODBC Cursor.
Count Distinct with Partitions	Determines how distinct counts of values are retrieved from partitioned tables.	• Do not select distinct elements for each partition • Select distinct elements for each partition	Do not select distinct elements for each partition

Property	Description	Possible Values	Default Value
Custom Group Banding Count Method	Helps optimize custom group banding when using the Count Banding method. You can choose to use the standard method that uses the Analytical Engine or database-specific syntax, or you can choose to use case statements or temp tables.	• Treat banding as normal calculation • Use standard case statement syntax • Insert band range to database and join with metric value	Treat banding as normal calculation
Custom Group Banding Points Method	Helps optimize custom group banding when using the Points Banding method. You can choose to use the standard method that uses the Analytical Engine or database-specific syntax, or you can choose to use case statements or temp tables.	• Treat banding as normal calculation • Use standard case statement syntax • Insert band range to database and join with metric value	Treat banding as normal calculation
Custom Group Banding Size Method	Helps optimize custom group banding when using the Size Banding method. You can choose to use the standard method that uses the Analytical Engine or database-specific syntax, or you can choose to use case statements or temp tables.	• Treat banding as normal calculation • Use standard case statement syntax • Insert band range to database and join with metric value	Treat banding as normal calculation

Property	Description	Possible Values	Default Value
Data population for Intelligent Cubes	Defines if and how Intelligent Cube data is normalized to save memory resources.	• Do not normalize Intelligent Cube data • Normalize Intelligent Cube data in Intelligence Server • Normalize Intelligent Cube data in database using Intermediate Table Type • Normalize Intelligent Cube data in database using Fallback Type • Normalize Intelligent Cube data basing on dimensions with attribute lookup filtering • Normalize Intelligent Cube data basing on dimensions with no attribute lookup filtering	Normalize Intelligent Cube data in Intelligence Server
Data population for reports	Defines if and how report data is normalized to save memory resources.	• Do not normalize report data • Normalize report data in Intelligence Server • Normalize report data in database using Intermediate Table Type • Normalize report data in database using Fallback Table Type • Normalize report data basing on dimensions with attribute lookup filtering	Do not normalize report data
Default Sort Behavior for Attribute Elements in Reports	Determines whether the sort order of attribute elements on reports considers special sort order formatting defined for attributes.	• Sort attribute elements based on the attribute ID form for each attribute • Sort attribute elements based on the defined 'Report Sort' setting of all attribute forms for each attribute	Sort attribute elements based on the attribute ID form for each attribute
Dimensionality Model	Determines level (dimension) replacement for non	• Use relational model • Use dimensional model	Use relational model

Property	Description	Possible Values	Default Value
	parent-child related attributes in the same hierarchy.		
Engine Attribute Role Options	Enable or disable the Analytical Engine's ability to treat attributes defined on the same column with the same expression as attribute roles.	• Enable Engine Attribute Role feature • Disable Engine Attribute Role feature	Disable Engine Attribute Role feature
Filter tree optimization for metric qualifications	Determines if metric qualifications that are included in separate passes of SQL are included in a single pass of SQL when possible.	• Enable Filter tree optimization for metric qualifications • Disable Filter tree optimization for metric qualifications	Enable Filter tree optimization for metric qualifications
Incremental Data Transfer	Determines whether data that is transferred between Intelligence Server and a database is performed using a single transfer of data or multiple, incremental transfers of data	• Enable Incremental Data Transfer • Disable Incremental Data Transfer	Disable Incremental Data Transfer
Maximum Parallel Queries Per Report	Determines how many queries can be executed in parallel as part of parallel query execution support	User-defined	2
MD Partition Prequery Option	Allows you to choose how to handle prequerying the metadata partition.	• Use count(*) in prequery • Use constant in prequery	Use count(*) in prequery
Multiple data source support	Defines which technique to use to support multiple data sources in a project.	• Use MultiSource Option to access multiple data sources • Use database gateway support to access multiple data sources	Use MultiSource Option to access multiple data source
OLAP function support	Defines whether OLAP functions support backwards compatibility or reflect enhancements to OLAP function logic.	• Preserve backwards compatibility with 8.1.x and earlier • Recommended with 9.0 and later	Preserve backwards compatibility with 8.1.x and earlier
Parallel Query	Determines whether	• Disable parallel query	Disable parallel query

Property	Description	Possible Values	Default Value
Execution	MicroStrategy attempts to execute multiple queries in parallel to return report results faster and publish Intelligent Cubes.	execution • Enable parallel query execution for multiple data source reports only • Enable parallel query execution for all reports that support it	execution
Parallel Query Execution Improvement Estimate in SQL View	Determines whether reports and Intelligent Cubes include an estimate in the percent of processing time that would be saved if parallel Query execution was used to run multiple queries in parallel.	• Disable parallel query execution improvement estimate in SQL view • Enable parallel query execution improvement estimate in SQL view	Disable parallel query execution improvement estimate in SQL view
Rank Method if DB Ranking Not Used	Determines how calculation ranking is performed.	• Use ODBC ranking (MSTR 6 method) • Analytical engine performs rank	Use ODBC ranking (MSTR 6 method).
Remove Aggregation Method	Determines whether to keep or remove aggregations in SQL queries executed from MicroStrategy.	• Remove aggregation according to key of FROM clause • Remove aggregation according to key of fact tables (old behavior)	Remove aggregation according to key of FROM clause
Remove Group By Option	Determines whether `Group By` and aggregations are used for attributes with the same primary key.	• Remove aggregation and Group By when Select level is identical to From level • Remove aggregation and Group By when Select level contains all attribute (s) in From level	Remove aggregation and Group By when Select level is identical to From level
Remove Repeated Tables For Outer Joins	Determines whether an optimization for outer join processing is enabled or disabled.	• Disable optimization to remove repeated tables in full outer join and left outer join passes • Enable optimization to remove repeated tables in full outer join and left outer join passes	Enable optimization to remove repeated tables in full outer join and left outer join passes
Set Operator Optimization	Allows you to use set operators in sub	• Disable Set Operator Optimization	Disable Set Operator Optimization

Property	Description	Possible Values	Default Value
	queries to combine multiple filter qualifications. Set operators are only supported by certain database platforms and with certain sub query types.	• Enable Set Operator Optimization (if supported by database and [Sub Query Type])	
SQL Global Optimization	Determines the level by which SQL queries in reports are optimized.	• Level 0: No optimization • Level 1: Remove Unused and Duplicate Passes • Level 2: Level 1 + Merge Passes with Different SELECT • Level 3: Level 2 + Merge Passes, which only hit DB Tables, with different WHERE • Level 4: Level 2 + Merge All Passes with Different WHERE	Level 4: Level 2 + Merge All Passes with Different WHERE
Sub Query Type	Allows you to determine the type of subquery used in engine-generated SQL.	• WHERE EXISTS (SELECT * ...) • WHERE EXISTS (SELECT col1, col2...) • WHERE COL1 IN (SELECT s1.COL1...) falling back to EXISTS (SELECT * ...) for multiple columns IN • WHERE (COL1, COL2...) IN (SELECT s1.COL1, s1.COL2...) • Use Temporary Table, falling back to EXISTS (SELECT *...) for correlated subquery • WHERE COL1 IN (SELECT s1.COL1...) falling back to EXISTS (SELECT col1, col2 ...) for multiple columns IN • Use Temporary Table, falling back to IN (SELECT COL) for correlated subquery	Use Temporary Table, falling back to EXISTS (SELECT *...) for correlated subquery

Property	Description	Possible Values	Default Value
Transformation Formula Optimization	Defines whether to attempt to improve performance of reports that use expression-based transformations.	• Always join with transformation table to perform transformation • Use transformation formula instead of join with transformation table when possible	Use transformation formula instead of join with transformation table when possible
Unrelated Filter Options	Determines whether the Analytical Engine should keep or remove the unrelated filter.	• Remove unrelated filter • Keep unrelated filter • Keep unrelated filter and put condition from unrelated attributes in one subquery group	Remove unrelated filter
Unrelated Filter Options for Nested Metrics	Determines whether the Analytical Engine should keep or remove the unrelated filters when using nested metrics.	• Use the 8.1.x behavior: • Use the 9.0.x behavior:	Use the 8.1.x behavior
WHERE Clause Driving Table	Determines the table used for qualifications in the WHERE clause.	• Use lookup table • Use fact table	Use fact table

Additional Final Pass Option

Additional Final Pass Option is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Additional Final Pass Option determines whether the Engine calculates an aggregation function and a join in a single pass or in separate passes in the SQL.

Levels at which you can set this

Report, template, and database instance

It is recommended that you use this property on reports. You must update the metadata to see the property populated in the metadata.

Example

The following SQL example was created using SQL Server metadata and warehouse.

Consider the following structure of lookup and fact tables:

- LU_Emp_Mgr has 4 columns, namely: Emp_ID, Emp_Desc, Mgr_ID, and Mgr_Desc

In this structure, Emp_ID is the primary key of LU_Emp_Mgr table

- LU_Dept has 2 columns, namely: Dept_ID and Dept_Desc

In this structure, Dept_ID is the primary key of LU_Dept table

- Fact table Emp_Dept_Salary has 3 columns, namely: Emp_ID, Dept_ID, and fact Salary

From the above warehouse structure, define the following schema objects:

- attribute Employee with 2 forms: Employee@ID (defined on column Emp_ID) and Employee@Desc (defined on column Emp_Desc)
- attribute Manager with 2 forms: Manager@ID (defined on column Mgr_ID) and Manager@Desc (defined on column Mgr_Desc)
- attribute Department with 2 forms: Department@ID (defined on column Dept_ID) and Department@Desc (defined on column Dept_Desc)
- fact Fact_Salary, which is defined on Salary column

The Manager attribute is defined as the parent of the Employee attribute via LU_Emp_Mgr table. This is a common practice in a star schema.

Create two metrics that are defined as

- Salary_Dept = Sum(Fact_Salary)){~+, Department+}
- Salary = Avg(Salary_Dept){~+}

In a report called Employee_Salary, put the Salarymetric on a template with the Managerattribute. In this example, the Employee_Salary report generates the following SQL:

```
Pass0
select a12.Mgr_Id  Mgr_Id,
       a11.Dept_Id  Dept_Id,
       sum(a11.Salary)  WJXBFS1
into #ZZTUW0200LXMD000
from dbo.Emp_Dept_Salary a11
       join dbo.Emp_Mgr a12
         on (a11.Emp_Id = a12.Emp_Id)
group by a12.Mgr_Id,
       a11.Dept_Id
Pass1
select pa1.Mgr_Id  Mgr_Id,
       max(a11.Mgr_Desc)  Mgr_Desc,
         avg(pa1.WJXBFS1)  WJXBFS1
from #ZZTUW0200LXMD000 pa1
         join dbo.Emp_Mgr a11
           on (pa1.Mgr_Id = a11.Mgr_Id)
group by pa1.Mgr_Id
Pass2
drop table #ZZTUW0200LXMD000
```

The problem in the SQL pass above that appears in italics is that the join condition and the aggregation function are in a single pass. The SQL joins the ZZTUW0200LXMD000 table to the Emp_Mgr table on column Mgr_ID, but Mgr_ID is not the primary key to the LU_Emp_Mgr table. Therefore, there are many rows on the LU_Emp_Mgr table with the same Mgr_ID. This results in a repeated data problem.

Clearly, if both the conditions, aggregation and join, do not exist on the same table, this problem does not occur.

To resolve this problem, select the option **One additional final pass only to join lookup tables** in the VLDB Properties Editor. With this option selected, the report, when executed, generates the following SQL:

```
Pass0
select a12.Mgr_Id  Mgr_Id,
       a11.Dept_Id  Dept_Id,
       sum(a11.Salary)  WJXBFS1
into #ZZTUW01006IMD000
from dbo.Emp_Dept_Salary a11
       join dbo.Emp_Mgr a12
         on (a11.Emp_Id = a12.Emp_Id)
group by a12.Mgr_Id,
       a11.Dept_Id
Pass1
select pa1.Mgr_Id  Mgr_Id,
         avg(pa1.WJXBFS1)  WJXBFS1
into #ZZTUW01006IEA001
from #ZZTUW01006IMD000 pa1
group by pa1.Mgr_Id
Pass2
select distinct pa2.Mgr_Id  Mgr_Id,
       a11.Mgr_Desc  Mgr_Desc,
       pa2.WJXBFS1  WJXBFS1
from #ZZTUW01006IEA001 pa2
         join dbo.Emp_Mgr a11
           on (pa2.Mgr_Id = a11.Mgr_Id)
Pass3
drop table #ZZTUW01006IMD000
Pass4
drop table #ZZTUW01006IEA001
```

In this SQL, the italicized sections show that the Engine calculates the aggregation function, which is the Average function, in a separate pass and performs the join operation in another pass.

Apply Filter Options

The Apply Filter property has three settings. The common element of all three settings is that report filters must be applied whenever a warehouse table is accessed. The settings are

- **Apply filter only to passes touching warehouse tables** (default): This is the default option. It applies the filter to only SQL passes that touch warehouse tables, but not to other passes. This option works in most situations.
- **Apply filter to passes touching warehouse tables and last join pass, if it does a downward join from the temporary table level to the template**

level: The filter is applied in the final pass if it is a downward join. For example, you have Store, Region Sales, and Region Cost on the report, with the filter "store=1." The intermediate passes calculate the total sales and cost for Region 1 (to which Store 1 belongs). In the final pass, a downward join is done from the Region level to the Store level, using the relationship table LOOKUP_STORE. If the "store = 1" filter in this pass is not applied, stores that belong to Region 1 are included on the report. However, you usually expect to see only Store 1 when you use the filter "store=1." So, in this situation, you should choose this option to make sure the filter is applied in the final pass.

- **Apply filter to passes touching warehouse tables and last join pass**: The filter in the final pass is always applied, even though it is not a downward join. This option should be used for special types of data modeling. For example, you have Region, Store Sales, and Store Cost on the report, with the filter "Year=2002." This looks like a normal report and the final pass joins from Store to Region level. But the schema is abnormal: certain stores do not always belong to the same region, perhaps due to rezoning. For example, Store 1 belongs to Region 1 in 2002, and belongs to Region 2 in 2003. To solve this problem, put an additional column Year in LOOKUP_STORE so that you have the following data.

Store	Region	Year
1	1	2002
1	2	2003
...		

Apply the filter Year=2002 to your report. This filter must be applied in the final pass to find the correct store-region relationship, even though the final pass is a normal join instead of a downward join.

Interaction with other VLDB properties

Two other VLDB properties, Downward Outer Join Option and Preserve All Lookup Table Elements, have an option to apply the filter. If you choose those options, then the filter is applied accordingly, regardless of what the value of Apply Filter Option is.

Levels at which you can set this

Database instance, report, and template

Attribute Element Number Count Method

Attribute Element Number Count Method is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The incremental fetch feature uses a SELECT COUNT DISTINCT query, introduced in MicroStrategy version 7.1.6. In some cases, this query can be costly for the data

warehouse and make the element browse time longer than necessary for certain production environments.

To alleviate this problem, the Attribute Element Number Count Method controls how the total number of rows are calculated. You have the following options:

- **Use Count(Attribute@ID) to calculate total element number (uses count distinct if necessary)** (default): In this case, the database determines the total number of rows.
- **Use ODBC cursor to calculate the total element number**: This setting causes Intelligence Server to determine the total number of rows by looping through the table after the initial SELECT pass.

The difference between the two approaches is whether the database or Intelligence Server determines the total number of records. MicroStrategy recommends using the "Use ODBC cursor..." option (having Intelligence Server determine the total number of records) if you have a heavily taxed data warehouse or if the SELECT COUNT DISTINCT query itself introduces contention in the database. Having Intelligence Server determine the total number of rows results in more traffic between Intelligence Server and the database.

For Tandem databases, the default is Use ODBC Cursor to calculate the total element number.

Levels at which you can set this

Database instance only

Count Distinct with Partitions

Count Distinct with Partitions is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This property can help improve the performance of queries performed on multiple partitioned tables which return a distinct count of values. A distinct count of values allows you to return information such as how many distinct types of items were sold on a given day. You have the following options:

- **Do not select distinct elements for each partition** (default): To return a distinct count of values from multiple partition tables, the tables are first combined together as one large result table, and then the count distinct calculation is performed. While this returns the proper results, combining multiple tables into one table to perform the count distinct calculation can be a resource-intensive query.
- **Select distinct elements for each partition**: To return a distinct count of values from multiple partitioned tables, the size of each partition table is first reduced by returning only distinct values. These smaller tables are then combined and a count distinct calculation is performed.

  This can improve performance by reducing the size of the partition tables before they are combined for the final count distinct calculation.

Levels at which you can set this

Metric, report, template, and database instance

Custom Group Banding Count Method

Custom Group Banding Count Method is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Custom Group Banding Count Method helps optimize custom group banding when using the Count Banding method. You have the following options:

- **Treat banding as normal calculation** (default): Select this option to allow the MicroStrategy Analytical Engine to perform the custom group banding.
- **Use standard case statement syntax**: Select this option to utilize case statements within your database to perform the custom group banding.
- **Insert band range to database and join with metric value**: Select this option to use temporary tables to perform the custom group banding.

Levels at which you can set this

Database instance, report, and template

Examples

The following SQL examples were created in MicroStrategy Tutorial. The report contains a Custom Group "Customer Value Banding", which uses the Count method and the Revenue metric. The SQL for each of the three settings for this property is presented below. All three options start with the same SQL passes. In this example, the first six passes are the same. The remaining SQL passes differ depending on the Custom Group Banding Count Method setting selected.

```
create table ZZMD00 (CUSTOMER_ID SHORT, WJXBFS1 DOUBLE)
insert into ZZMD00
select a11.CUSTOMER_ID AS CUSTOMER_ID, a11.TOT_DOLLAR_SALES
                as WJXBFS1
from CUSTOMER_SLS a11
create table ZZMD01 (WJXBFS1 DOUBLE)
insert into ZZMD01
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from YR_CATEGORY_SLS a11
select pa1.CUSTOMER_ID AS CUSTOMER_ID,
                (pa1.WJXBFS1 / pa2.WJXBFS1) as WJXBFS1
from ZZMD00 pa1, ZZMD01 pa2
create table ZZMQ02 (CUSTOMER_ID SHORT, DA57 LONG)
[Placeholder for an analytical SQL]
insert into ZZMQ02 values (DummyInsertValue)
```

Treat banding as normal calculation (default)

```
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from CUSTOMER_SLS a11, ZZMQ02 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
select a12.DA57 AS DA57, sum(a11.TOT_DOLLAR_SALES)
```

```
                as WJXBFS1
from CUSTOMER_SLS a11, ZZMQ02 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
group by a12.DA57
drop table ZZMD00
drop table ZZMD01
drop table ZZMQ02
```

Use standard case statement syntax

```
create table ZZOP03 (CUSTOMER_ID SHORT, DA57 LONG)
insert into ZZOP03
select pa3.CUSTOMER_ID AS CUSTOMER_ID,
 (case
        when (pa3.WJXBFS1 >=1 and pa3.WJXBFS1 < 100.9)
                then 1
        when (pa3.WJXBFS1 >= 100.9 and pa3.WJXBFS1 200.8)
                then 2
        when (pa3.WJXBFS1 >= 200.8 and pa3.WJXBFS1 300.7)
                then 3
        when (pa3.WJXBFS1 >= 300.7 and pa3.WJXBFS1 400.6)
                then 4
        when (pa3.WJXBFS1 >= 400.6 and pa3.WJXBFS1 500.5)
                then 5
        when (pa3.WJXBFS1 >= 500.5 and pa3.WJXBFS1 600.4)
                then 6
        when (pa3.WJXBFS1 >= 600.4 and pa3.WJXBFS1 700.3)
                then 7
        when (pa3.WJXBFS1 >= 700.3 and pa3.WJXBFS1 800.2)
                then 8
        when (pa3.WJXBFS1 >= 800.2 and pa3.WJXBFS1 900.1)
                then 9
        when (pa3.WJXBFS1 >= 900.1 and pa3.WJXBFS1 <= 1000
                then 10
 end) as DA57
from ZZMQ02 pa3
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from CUSTOMER_SLS a11, ZZOP03 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
select a12.DA57 AS DA57, sum(a11.TOT_DOLLAR_SALES)
                as WJXBFS1
from CUSTOMER_SLS a11, ZZOP03 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
group by a12.DA57
drop table ZZMD00
drop table ZZMD01
drop table ZZMQ02
drop table ZZOP03
```

Insert band range to database and join with metric value

```
create table ZZOP03 (BandNo LONG, BandStart DOUBLE,
                BandEnd DOUBLE)
insert into ZZOP03 values (1, 1, 100.9)
[Insertions for other bands]
create table ZZOP04 (CUSTOMER_ID SHORT, DA57 LONG)
insert into ZZOP04
select pa3.CUSTOMER_ID AS CUSTOMER_ID, pa4.BandNo as DA57
from ZZMQ02 pa3, ZZOP03 pa4
where ((pa3.WJXBFS1 >= pa4.BandStart
        and pa3.WJXBFS1 < pa4.BandEnd)
         or (pa3.WJXBFS1 = pa4.BandEnd
        and pa4.BandNo = 10))
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from CUSTOMER_SLS a11, ZZOP04 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
```

```
select a12.DA57 AS DA57, sum(a11.TOT_DOLLAR_SALES)
                as WJXBFS1
from CUSTOMER_SLS a11, ZZOP04 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
group by a12.DA57
drop table ZZMD00
drop table ZZMD01
drop table ZZMQ02
drop table ZZOP03
drop table ZZOP04
```

Custom Group Banding Points Method

Custom Group Banding Point Method is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Custom Group Banding Points Method helps optimize custom group banding when using the Points Banding method. You can choose to use the standard method that uses the Analytical Engine or database-specific syntax, or you can choose to use case statements or temp tables.

Levels at which you can set this

Database instance, report, and template

Examples

The following SQL examples were created in MicroStrategy Tutorial. The report contains a Custom Group "Customer Value Banding" using the point method and the Revenue metric. The SQL for each of the three settings for this property is presented below. All three options start with the same SQL passes. In this example, the first six passes are the same. The remaining SQL passes differ depending on the Custom Group Banding Count Method selected.

```
create table ZZMD00 (CUSTOMER_ID SHORT, WJXBFS1 DOUBLE)
insert into ZZMD00
select a11.CUSTOMER_ID AS CUSTOMER_ID, a11.TOT_DOLLAR_SALES
                as WJXBFS1
from CUSTOMER_SLS a11
create table ZZMD01 (WJXBFS1 DOUBLE)
insert into ZZMD01
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from YR_CATEGORY_SLS a11
select pa1.CUSTOMER_ID AS CUSTOMER_ID,
                (pa1.WJXBFS1 / pa2.WJXBFS1) as WJXBFS1
from ZZMD00 pa1, ZZMD01 pa2
create table ZZMQ02 (CUSTOMER_ID SHORT, DA57 LONG)
[Placeholder for an analytical SQL]
insert into ZZMQ02 values (DummyInsertValue)
```

Treat banding as normal calculation (default)

```
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from CUSTOMER_SLS a11, ZZMQ02 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
select a12.DA57 AS DA57, sum(a11.TOT_DOLLAR_SALES)
                as WJXBFS1
```

```
from CUSTOMER_SLS a11, ZZMQ02 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
group by a12.DA57
drop table ZZMD00
drop table ZZMD01
drop table ZZMQ02
Use standard case statement syntax
create table ZZOP03 (CUSTOMER_ID SHORT, DA57 LONG)
insert into ZZOP03
select pa3.CUSTOMER_ID AS CUSTOMER_ID,
(case
        when (pa3.WJXBFS1 >= 1 and pa3.WJXBFS1 < 2) then 1
        when (pa3.WJXBFS1 >= 2 and pa3.WJXBFS1 <= 3)then 2
end) as DA57
from ZZMQ02 pa3
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from CUSTOMER_SLS a11, ZZOP03 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
select a12.DA57 AS DA57, sum(a11.TOT_DOLLAR_SALES)
                as WJXBFS1
from CUSTOMER_SLS a11, ZZOP03 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
group by a12.DA57
drop table ZZMD00
drop table ZZMD01
drop table ZZMQ02
drop table ZZOP03
```

Insert band range to database and join with metric value

```
create table ZZOP03 (BandNo LONG, BandStart DOUBLE,
                BandEnd DOUBLE)
insert into ZZOP03 values (1, 1, 2)
[Insertions for other bands]
create table ZZOP04 (CUSTOMER_ID SHORT, DA57 LONG)
insert into ZZOP04
select pa3.CUSTOMER_ID AS CUSTOMER_ID, pa4.BandNo as DA57
from ZZMQ02 pa3, ZZOP03 pa4
where ((pa3.WJXBFS1 >= pa4.BandStart
        and pa3.WJXBFS1 < pa4.BandEnd)
         or (pa3.WJXBFS1 = pa4.BandEnd
        and pa4.BandNo = 2))
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from CUSTOMER_SLS a11, ZZOP04 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
select a12.DA57 AS DA57, sum(a11.TOT_DOLLAR_SALES)
                as WJXBFS1
from CUSTOMER_SLS a11, ZZOP04 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
group by a12.DA57
drop table ZZMD00
drop table ZZMD01
drop table ZZMQ02
drop table ZZOP03
drop table ZZOP04
```

Custom Group Banding Size Method

Custom Group Banding Size Method is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Custom Group Banding Size Method helps optimize custom group banding when using the Size Banding method. You can choose to use the standard method that uses the Analytical Engine or database-specific syntax, or you can choose to use case statements or temp tables.

Levels at which you can set this

Database instance, report, and template

Examples

The following SQL examples were created in MicroStrategy Tutorial. The report contains a Custom Group "Customer Value Banding" that uses the size method and the Revenue metric. The SQL for each of the three settings for this property is presented below. All three options start with the same SQL passes. In this example, the first six passes are the same. The remaining SQL passes differ depending on the Custom Group Banding Count Method selected.

```
create table ZZMD000 (CUSTOMER_ID SHORT, WJXBFS1 DOUBLE)
insert into ZZMD000
select a11.CUSTOMER_ID AS CUSTOMER_ID, a11.TOT_DOLLAR_SALES
                as WJXBFS1
from CUSTOMER_SLS a11
create table ZZMD001 (WJXBFS1 DOUBLE)
insert into ZZMD001
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from YR_CATEGORY_SLS a11
select pa1.CUSTOMER_ID AS CUSTOMER_ID,
                (pa1.WJXBFS1 / pa2.WJXBFS1) as WJXBFS1
from ZZMD000 pa1, ZZMD001 pa2
create table ZZMQ002 (CUSTOMER_ID SHORT, WJXBFS1 DOUBLE)
 [Placeholder for an Analytical SQL]
insert into ZZMQ02 values (DummyInsertValue)
```

Treat banding as normal calculation

```
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from CUSTOMER_SLS a11, ZZMQ002 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
select a12.DA57 AS DA57, sum(a11.TOT_DOLLAR_SALES)
                as WJXBFS1
from CUSTOMER_SLS a11, ZZMQ002 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
group by a12.DA57
drop table ZZMD000
drop table ZZMD001
drop table ZZMQ002
```

Use standard CASE statement syntax

```
create table ZZOP003 (CUSTOMER_ID SHORT, DA57 LONG)
insert into ZZOP003
select pa3.CUSTOMER_ID AS CUSTOMER_ID,
 (case
        when (pa3.WJXBFS1 >= 0 and pa3.WJXBFS1 < .2) then 1
        when (pa3.WJXBFS1 >= .2 and pa3.WJXBFS1 < .4)then 2
        when (pa3.WJXBFS1 >= .4 and pa3.WJXBFS1 < .6)then 3
        when (pa3.WJXBFS1 >= .6 and pa3.WJXBFS1 < .8)then 4
        when (pa3.WJXBFS1 >= .8 and pa3.WJXBFS1 <= 1)then 5
 end) as DA57
from ZZMQ002 pa3
```

```
drop table ZZMD000
drop table ZZMD001
drop table ZZMQ002
drop table ZZOP003
```

Insert band range to database and join with metric value

```
create table ZZOP003 (BandNo LONG, BandStart DOUBLE,
                BandEnd DOUBLE)
insert into ZZOP003 values (1, 0, .2)
[Insertions for other bands]
create table ZZOP004 (
        CUSTOMER_ID SHORT,
        DA57 LONG)
insert into ZZOP004
select pa3.CUSTOMER_ID AS CUSTOMER_ID, pa4.BandNo as DA57
from ZZMQ002 pa3, ZZOP003 pa4
where ((pa3.WJXBFS1 >= pa4.BandStart
        and pa3.WJXBFS1 < pa4.BandEnd)
         or (pa3.WJXBFS1 = pa4.BandEnd
        and pa4.BandNo = 5))
select sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from CUSTOMER_SLS a11, ZZOP004 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
select a12.DA57 AS DA57,sum(a11.TOT_DOLLAR_SALES) as WJXBFS1
from CUSTOMER_SLS a11, ZZOP004 a12
where a11.CUSTOMER_ID = a12.CUSTOMER_ID
group by a12.DA57
drop table ZZMD000
drop table ZZMD001
drop table ZZMQ002
drop table ZZOP003
drop table ZZOP004
```

Data population for Intelligent Cubes

The Data population for Intelligent Cubes VLDB property allows you to define if and how Intelligent Cube data is normalized to save memory resources.

When an Intelligent Cube is published, the description information for the attributes (all data mapped to non-ID attribute forms) included on the Intelligent Cube is repeated for every row. For example, an Intelligent Cube includes the attributes Region and Store, with each region having one or more stores. Without performing normalization, the description information for the Region attribute would be repeated for every store. If the South region included five stores, then the information for South would be repeated five times.

You can avoid this duplication of data by normalizing the Intelligent Cube data. In this scenario, the South region description information would only be stored once even though the region contains five stores. While this saves memory resources, the act of normalization requires some processing time. This VLDB property provides the following options to determine if and how Intelligent Cube data is normalized:

- **Do not normalize Intelligent Cube data**: Intelligent Cube data is not normalized. The memory resources required for the Intelligent Cube may be far greater than if one of the other normalization options is performed. This option is best suited for troubleshooting purposes only.

- **Normalize Intelligent Cube data in Intelligence Server** (default): Intelligence Server performs the Intelligent Cube data normalization. This typically processes the normalization faster than the other normalization options, but also requires memory resources of Intelligence Server.

  This is a good option if you publish your Intelligent Cubes at times when Intelligence Server use is low. Normalization can then be performed without affecting your user community. You can use schedules to support this strategy. For information on using schedules to publish Intelligent Cubes, see the *In-memory Analytics Guide*.

- The other options available for Intelligent Cube normalization all perform the normalization within the database. Therefore, these are all good options if Intelligent Cubes are published when Intelligence Server is in use by the user community, or any time when the memory resources of Intelligence Server must be conserved.

  You can see improved performance with the database normalization techniques if the Intelligent Cube is retrieving a large ratio of repeating data. However, normalizing data within the database is typically slower than normalizing the data in Intelligence Server.

  Each database normalization technique is described below:

  - **Normalize Intelligent Cube data in database using Intermediate Table Type**: This option is no longer available. If you upgraded a project from version 9.0.0 and this option was in use, this option is still used until you manually select a different option. Once you select a different option, you cannot revert to the behavior for this option.

    If you used this option in 9.0.0 and have upgraded to the most recent version of MicroStrategy, it is recommended that you use a different Intelligent Cube normalization technique. If the user account for the data warehouse has permissions to create tables, switch to the option **Normalize Intelligent Cube data in the database**. This option is described below. If the user account does not have permissions to create tables, switch to the option **Normalize Intelligent Cube data in Intelligence Server**.

  - **Normalize Intelligent Cube data in the database**: This database normalization is a good option if attribute data and fact data are stored in the same table.

    To use this option, the user account for the database must have permissions to create tables.

  - **Normalize Intelligent Cube data in the database using relationship tables**: This database normalization is a good option if attribute data and fact data are stored in separate tables.

    To use this option, the user account for the database must have permissions to create tables.

  - **Direct loading of dimensional data and filtered fact data**: This database normalization is a good option if attribute data and fact data are stored in separate tables, and the Intelligent Cube includes the majority of the attribute elements for each attribute it uses.

This is a resource-intensive option, and for very large Intelligent Cubes, enabling this setting may deplete your Intelligence Server's system memory.

To use this option, the user account for the database must have permissions to create tables. Additionally, using this option can return different results than the other Intelligent Cube normalization techniques. For information on these differences, see *Data differences when normalizing Intelligent Cube data using direct loading, page 153* below.

Data differences when normalizing Intelligent Cube data using direct loading

The option **Direct loading of dimensional data and filtered fact data** can return different results than the other Intelligent Cube normalization techniques in certain scenarios. Some of these scenarios and the effect that they have on using direct loading for Intelligent Cube normalization are described below:

- There are extra rows of data in fact tables that are not available in the attribute lookup table. In this case the VLDB property Preserve all final pass result elements (see *Relating column data with SQL: Joins, page 59*) determines how to process the data. The only difference between direct loading and the other normalization options is that the option Preserve all final result pass elements and the option Preserve all elements of final pass result table with respect to lookup table but not relationship table both preserve the extra rows by adding them to the lookup table.
- There are extra rows of data in the attribute lookup tables that are not available in the fact tables. With direct loading, these extra rows are included. For other normalization techniques, the VLDB property Preserve all lookup table elements (see *Relating column data with SQL: Joins, page 59*) determines whether or not to include these rows.
- The Intelligent Cube includes metrics that use OLAP functions. If an Intelligent Cube includes metrics that use OLAP functions, you should use an Intelligent Cube normalization technique other than the direct loading technique to ensure that the data returned is accurate.

  OLAP functions are functions such as `RunningSum`, `MovingAvg`, and `OLAPMax`. For information about how to use OLAP functions, see the Functions Reference.

Levels at which you can set this

Database instance, report, and template

Data population for reports

The Data population for reports VLDB property allows you to define if and how report data is normalized to save memory resources.

When a report is executed, the description information for the attributes (all data mapped to non-ID attribute forms) included on the report is repeated for every row. For example, a report includes the attributes Region and Store, with each region having one or more stores. Without performing normalization, the description information for the

Region attribute would be repeated for every store. If the South region included five stores, then the information for South would be repeated five times.

You can avoid this duplication of data by normalizing the report data. In this scenario, the South region description information would only be stored once even though the region contains five stores. While this saves memory resources, the act of normalization requires some processing time. This VLDB property provides the following options to determine if and how report data is normalized:

- **Do not normalize report data** (default): Report data is not normalized. While no extra processing is required to normalize the report data, the memory resources required for the report are larger than if normalization was performed. However, reports commonly do not return large result sets and thus do not suffer from performance issues related to this duplication of data. Therefore, this option is the default for all reports.
- **Normalize report data in Intelligence Server**: Intelligence Server performs the report data normalization. This typically processes the normalization faster than the other normalization options, but also requires memory resources of Intelligence Server. This is a good option if report performance is the top priority.
- The other options available for report data normalization all perform the normalization within the database. Therefore, these are all good options if the memory resources of Intelligence Server must be conserved.

  You can see improved performance with the database normalization techniques if the report is retrieving a large ratio of repeating data. However, normalizing data within the database is typically slower than normalizing the data in Intelligence Server.

  Each database normalization technique is described below:

  - **Normalize report data in database using Intermediate Table Type**: This option is no longer available. If you upgraded a project from version 9.0.0 and this option was in use, this option is still used until you manually select a different option. Once you select a different option, you cannot revert to the behavior for this option.

    If you used this option in 9.0.0 and have upgraded to the most recent version of MicroStrategy, it is recommended that you use a different report data normalization technique. If the user account for the data warehouse has permissions to create tables, switch to the option **Normalize report data in the database**. This option is described below. If the user account does not have permissions to create tables, switch to the option **Normalize report data in Intelligence Server**.
  - **Normalize report data in the database**: This database normalization is a good option if attribute data and fact data are stored in the same table.

    To use this option, the user account for the database must have permissions to create tables.
  - **Normalize report data in the database using relationship tables**: This database normalization is a good option if attribute data and fact data are stored in separate tables.

To use this option, the user account for the database must have permissions to create tables.

Levels at which you can set this

Database instance, report, and template

Default Sort Behavior for Attribute Elements in Reports

The Default Sort Behavior for Attribute Elements in Reports VLDB property determines whether the sort order of attribute elements on reports considers special sort order formatting defined for attributes:

- **Sort attribute elements based on the attribute ID form for each attribute (default)**: Reports automatically use the attribute ID form to sort the results of a report. This is the default behavior seen prior to defining any sorting options for the report. If you define a report to use the default advanced sorting for an attribute, any sorting defined for the attribute forms of the attribute are then applied to the report. For information on advanced sorting for reports, see the Advanced Reporting Guide.
- **Sort attribute elements based on the defined 'Report Sort' setting of all attribute forms for each attribute**: Reports automatically use any sorting defined for the attribute forms of the attribute. In this scenario, no advanced sorting needs to be defined for a report to consider any attribute form sorting defined for the attributes on the report.

  An example of where this option can be helpful is when an attribute has an attribute form that is used solely for sorting the elements of an attribute on a report. An attribute form like this can be required if the ID values do not represent the order in which the attribute elements should be displayed by default on the report, and the specifics of the sort order are not relevant and therefore should not be displayed to report analysts. This sort order column can be added to the attribute, defined with a specific sort order for the attribute, and also defined to not be included as an available report form. By defining the attribute form in this way and selecting this VLDB property option, the attribute form is not displayed on reports, but it is still used to automatically sort the values of the report without having to define any advanced sorting for the report.

Levels at which you can set this

Database instance only

Dimensionality Model

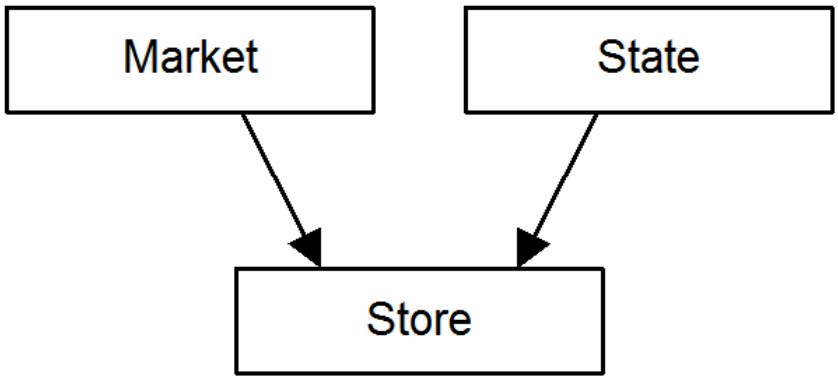
Dimensionality Model is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Dimensionality Model property is strictly for backward compatibility with MicroStrategy 6.x or earlier.

- **Use relational model** (default): For all projects, Use relational model is the default value. With the Use relational model setting, all the dimensionality (level) resolution is based on the relationship between attributes.
- **Use dimensional model**: The Use dimensional model setting is for cases where attribute relationship dimensionality (level) resolution is different from dimension-based resolution. There are very few cases when the setting needs to be changed to Use dimensional model. The following situations may require the Use dimensional model setting:
  - Metric Conditionality: You have a report with the Year attribute and the "Top 3 Stores Dollar Sales" metric on the template and the filters Store, Region, and Year. Therefore, the metric has a metric conditionality of "Top 3 Stores." In MicroStrategy 7.x and later, metric conditions are set to remove related report filter elements by default; therefore, with the above report the filters on Store and Region are ignored, because they are related to the metric conditionality. Year is not removed because it is not related to the metric conditionality. In MicroStrategy 7.x and later, if you set this property to Use dimensional model, the filters are not ignored.

    If you change the default of the Remove related report filter element option in advanced conditionality, the Use dimensional model setting does not make a difference in the report. For more information regarding this advanced setting, see the *Metrics* chapter in the *Advanced Reporting Guide*.

  - Metric Dimensionality Resolution: MicroStrategy 7.x and later does not have the concept of dimension, but instead has the concept of metric level. For a project upgraded from 6.x to 7.x, the dimension information is kept in the metadata. Attributes created in 7.x do not have this information. For example, you have a report that contains the Year attribute and the metric "Dollar Sales by Geography." The metric is defined with the dimensionality of Geography, which means the metric is calculated at the level of whatever Geography attribute is on the template. In MicroStrategy 7.x and later, the metric dimensionality is ignored and therefore defaults to the report level or the level that is defined for the report.
  - Analysis level calculation: For the next situation, consider the following split hierarchy model.



Market and State are both parents of Store. A report has the attributes Market and State and a Dollar Sales metric with report level dimensionality. In MicroStrategy 7.x and later, with the Use relational model setting, the report level (metric dimensionality level) is Market and State. To choose the best fact table to use to produce this report, the Analytical Engine considers both of these attributes. With the Use dimensional model setting in MicroStrategy 7.x and later, Store is used as the metric dimensionality level and for determining the best fact table to use. This is because Store is the highest common descendent between the two attributes.

Levels at which you can set this

Database instance, report, and template

Engine Attribute Role Options

Engine Attribute Role Options is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Engine Attribute Role Options property allows you to share an actual physical table to define multiple schema objects. There are two approaches for this feature:

- The first approach is a procedure called table aliasing, where you can define multiple logical tables in the schema that point to the same physical table, and then define different attributes and facts on these logical tables. Table aliasing provides you a little more control and is best when upgrading or when you have a complex schema. Table aliasing is described in detail in the *MicroStrategy Project Design Guide*.
- The second approach is called Engine Attribute Role. With this approach, rather than defining multiple logical tables, you only need to define multiple attributes and facts on the same table. The MicroStrategy Engine automatically detects "multiple roles" of certain attributes and splits the table into multiple tables internally. There is a limit on the number of tables into which a table can split. This limit is known as the Attribute Role limit. This limit is hard coded to 128 tables. If you are a new MicroStrategy user starting with *7i* or later, it is suggested that you use the automatic detection (Engine Attribute Role) option.

The algorithm to split the table is as follows:

- If two attributes are defined on the same column from the same table, have the same expression, and are not related, it is implied that they are playing different roles and must be in different tables after the split.
- If two attributes are related to each other, they must stay in the same table after the split.
- Attributes should be kept in as many tables as possible as long as algorithm 1 is not violated.

Given the diversity of data modeling in projects, the above algorithm cannot be guaranteed to split tables correctly in all situations. Thus, this property is added in the VLDB properties to turn the Engine Attribute Role on or off. When the feature is turned off, the table splitting procedure is bypassed.

Correct usage example

Fact table FT1 contains the columns "Order_Day," "Ship_Day," and "Fact_1." Lookup table LU_DAY has columns "Day," "Month," and "Year." Attributes "Ship Day" and "Order Day" are defined on different columns in FT1, but they share the same column ("Day") on LU_DAY. Also the attributes "Ship Month" and "Order Month" share the same column "month" in LU_DAY. The "Ship Year" and "Order Year" attributes are the same as well. During the schema loading, the Analytical Engine detects the duplicated definitions of attributes on column "Day," "Month," and "Year." It automatically splits LU_DAY into two internal tables, LU_DAY(1) and LU_DAY(2), both having the same physical table name LU_DAY. As a result, the attributes "Ship Day," "Ship Month," and "Ship Year" are defined on LU_DAY(1) and "Order Day," "Order Month," and "Order Year" are defined on LU_DAY(2). Such table splitting allows you to display Fact_1 that is ordered last year and shipped this year.

The SQL appears as follows:

```
select a1.fact_1
from FT1 a1 join LU_DAY a2 on (a1.order_day=a2.day)
        join LU_DAY a3 on (a1.ship_day = a3.day)
where a2.year = 2002 and
a3.year = 2003
```

Note that LU_DAY appears twice in the SQL, playing different "roles." Also, note that in this example, the Analytical Engine does not split table FT1 because "Ship Day" and "Order Day" are defined on different columns.

Incorrect usage example

Fact table FT1 contains columns "day" and "fact_1." "Ship Day" and "Order Day" are defined on column "day." The Analytical Engine detects that these two attributes are defined on the same column and therefore splits FT1 into FT1(1) and FT1(2), with FT1(1) containing "Ship Day" and "Fact 1", and FT(2) containing "Order Day" and "Fact 1." If you put "Ship Day" and "Order Day" on the template, as well as a metric calculating "Fact 1," the Analytical Engine cannot find such a fact. Although externally, FT1 contains all the necessary attributes and facts, internally, "Fact 1" only exists on either "Ship Day" or "Order Day," but not both. In this case, to make the report work (although still incorrectly), you should turn OFF the Engine Attribute Role feature.

- Because of backward compatibility and because the Analytical Engine's automatic splitting of tables may be wrong for some data models, this property's default setting is to turn OFF the Engine Attribute Role feature.
- If this property is turned ON, and you use this feature incorrectly, the most common error message from the Analytical Engine is

```
Fact not found at requested level.
```

- This feature is turned OFF by default starting from 7*i* Beta 2. Before that, this feature was turned OFF for upgraded projects and turned ON by default for new projects. So for some 7*i* beta users, if you create a new metadata using the Beta1 version of 7*i*, this feature may be turned on in your metadata.
- While updating the schema, if the Engine Attribute Role feature is ON, and if the Attribute Role limit is exceeded, you may get an error message from the Engine. You get this error because there is a limit on the number of tables into which a given table can be split internally. In this case, you should turn the Engine Attribute Role feature OFF and use table aliasing instead.

Levels at which you can set this

Database instance only

Filter tree optimization for metric qualifications

Filter tree optimization for metric qualifications is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Filter tree optimization for metric qualifications property determines whether metric qualifications that are included in separate passes of SQL are included in a single pass of SQL when possible. Metric qualifications can be included in separate passes of SQL in scenarios such as when the metric qualifications are used in filter definitions. Having a metric qualification at each logical level of a filter qualification can include each metric qualification in a separate pass of SQL. For example, consider a filter qualification that is structured as follows:

```
(AttributeQualfication1 AND MetricQualification1) AND
(AttributeQualification2 AND MetricQualification2)
```

Since `MetricQualification1` and `MetricQualification2` are at separate logical levels of the filter qualification, this can cause each metric qualification to require its own pass of SQL.

You have the following options for this VLDB property:

- **Enable Filter tree optimization for metric qualifications**: Defines metric qualifications to be included in the same pass of SQL if possible. In the scenario

described above, `MetricQualification1` and `MetricQualification2` are processed in the same pass of SQL. This can help to improve performance by reducing the number of SQL passes required.

- **Disable Filter tree optimization for metric qualifications**: Defines metric qualifications to be included in SQL passes based on their logical level in a filter qualification. In the scenario described above, `MetricQualification1` and `MetricQualification2` are processed in different passes of SQL.

Levels at which you can set this

Report and project

Incremental Data Transfer

The Incremental Data Transfer VLDB property determines whether data that is transferred between Intelligence Server and a data source is performed using a single transfer of data or multiple, incremental transfers of data. Transferring data between Intelligence Server and a data source can be required for MicroStrategy features such as MicroStrategy MultiSource Option, data marts, bulk export to export large reports as delimited text files, and other Analytical Engine features.

This VLDB property has the following options:

- **Enable Incremental Data Transfer**: Data that is transferred between Intelligence Server and a data source is transferred using multiple, incremental transfers of data. This can improve performance in scenarios where large amounts of data from Intelligence Server are written to a data source. By transferring the data incrementally, some data can be written to the data source while additional data is retrieved through Intelligence Server.
- **Disable Incremental Data Transfer** (default): Data that is transferred between Intelligence Server and a data source is transferred using a single transfer of data.

Levels at which you can set this

Project and report

Maximum Parallel Queries Per Report

Maximum Parallel Queries Per Report is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Maximum Parallel Queries Per Report property determines how many queries can be executed in parallel as part of parallel query execution support. By default, a maximum of two queries can be executed in parallel, and you can increase this number to perform additional queries in parallel. For data that is integrated into MicroStrategy using Data Import, the default maximum number of queries that can be executed in parallel is five. When determining this maximum, consider the following:

- You must enable parallel Query execution to perform multiple queries in parallel. To enable parallel Query execution, see *Maximum Parallel Queries Per Report, page 160*.
- The number of queries executed in parallel is also dependent on the report or Intelligent Cube that is being executed. For example, if the maximum is set to three but a report only uses two passes of SQL, then only those two queries can be performed in parallel.
- When multiple queries are executed in parallel, this means that the actual processing of the multiple queries is performed in parallel on the database. If a database is required to do too many tasks at the same time this can cause the response time of the database to slow down, and thus degrade the overall performance. You should take into account the databases used to retrieve data and their available resources when deciding how to restrict parallel Query execution.

Levels at which you can set this

Project only

MD Partition Prequery Option

The purpose of the MD Partition Prequery Option is to find out which partition base table is used. The report filter is combined with partition base table filters. If the intersection of both filters is not empty, then the corresponding partition base table should be used. A SELECT statement for each partition base table is generated, and the query result is checked to see whether it is empty.

There are multiple ways to generate a SELECT statement that checks for the data, but the performance of the query can differ depending on the platform. The default value for this property is: “select count(*) ...” for all database platforms, except UDB, which uses “select distinct 1...”

Levels at which you can set this

Database instance, report, and template

Multiple data source support

The Multiple data source support VLDB property allows you to choose which technique to use to support multiple data sources in a project. This VLDB property has the following options:

- **Use MultiSource Option to access multiple data sources** (default): MultiSource Option is used to access multiple data sources in a project.

  MicroStrategy includes an extension to Intelligence Server referred to as MultiSource Option. With this feature, you can connect a project to multiple relational data sources. This allows you to integrate all your information from various databases and other relational data sources into a single MicroStrategy project for reporting and

analysis purposes. All data sources included by using the MultiSource Option are integrated as part of the same relational schema for a project.

- **Use database gateway support to access multiple data sources**: Database gateways are used to access multiple data sources in a project.

  You can specify a secondary database instance for a table, which is used to support database gateways. For example, in your environment you might have a gateway between two databases such as an Oracle database and a DB2 database. One of them is the primary database and the other is the secondary database. The primary database receives all SQL requests and passes them to the correct database.

For more information on both techniques for connecting to multiple data sources, see the *Project Design Guide*.

Levels at which you can set this

Database instance and project

OLAP function support

The OLAP function support VLDB property defines whether OLAP functions support backwards compatibility or reflect enhancements to OLAP function logic. This VLDB property has the following options:

- **Preserve backwards compatibility with 8.1.x and earlier** (default): OLAP functions reflect the functionality of pre-9.0 releases to support backwards compatibility.

  This behavior does not correctly use multiple passes for nested or sibling metrics that use OLAP functions. It also does not correctly apply attributes in the SortBy and BreakBy parameters.

- **Recommended with 9.0 and later**: OLAP functions reflect the enhancements included in 9.0 and later releases.

  This recommended behavior uses multiple passes for nested or sibling metrics that use OLAP functions. The functions also ignore attributes in SortBy or BreakBy parameters when the attributes are children of or are unrelated to the component metric's level.

Levels at which you can set this

Database instance, report, and template

Parallel Query Execution

Parallel Query Execution is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Parallel Query Execution property determines whether MicroStrategy attempts to execute multiple queries in parallel to return report results faster and publish Intelligent Cubes. This VLDB property has the following options:

- **Disable parallel query execution** (default): All queries for MicroStrategy reports and Intelligent Cubes are processed sequentially.

  Disabling parallel query execution by default allows you to first verify that your reports and Intelligent Cubes are executing correctly prior to any parallel query optimization. If you enable parallel query execution and errors are encountered or data is not being returned as expected, disabling parallel query execution can help to troubleshoot the report or Intelligent Cube.

- **Enable parallel query execution for multiple data source reports only**: MicroStrategy attempts to execute multiple queries in parallel for MicroStrategy reports and Intelligent Cubes that access multiple data sources. You can access multiple data sources using either MicroStrategy MultiSource Option, or database gateway support. To enable one of these options, see *Parallel Query Execution, page 162*.

  For reports and Intelligent Cubes that do not use MultiSource Option or database gateway support to access multiple data sources, all queries are processed sequentially.

- **Enable parallel query execution for all reports that support it**: MicroStrategy attempts to execute multiple queries in parallel for all MicroStrategy reports and Intelligent Cubes. This option is automatically used for data that you integrate into MicroStrategy using Data Import.

How parallel query execution is supported

To support parallel query execution, MicroStrategy analyzes the query logic that will be run for a report or Intelligent Cube for potential multiple queries. Multiple queries are used for tasks that require:

- The creation of tables to store intermediate results, which are then used later in the same query.

  These intermediate results must be stored as permanent tables to be considered for parallel query execution. These permanent tables are required to ensure that the parallel query execution results are available for separate database sessions and connections. If database features including derived tables or common table expressions are used, parallel query execution cannot be used because these techniques are considered to be a single query, which cannot be divided into separate pieces. Therefore, data sources that use permanent tables to store intermediate results are good candidates for parallel query execution.

  MicroStrategy uses derived tables and common table expressions by default for databases that are well-suited to use these features to store intermediate results. These databases can often perform their own query optimizations using either derived tables or common table expressions, and therefore may be better suited to using these techniques rather than using MicroStrategy's parallel query execution.

- Selecting independent lookup, relationship, or fact data using SQL normalization or direct data loading methods. For information on using these techniques with

Intelligent Cubes and reports, see *Parallel Query Execution, page 162* and *Parallel Query Execution, page 162* respectively.

- Loading multiple tables imported using Data Import, to publish a dataset. The option Enable parallel query execution for all reports that support it is automatically used for data that you integrate into MicroStrategy using Data Import.

Candidates for parallel query execution

Simple reports in MicroStrategy may not require multiple queries to return the required results, so even if parallel query execution is enabled, there may be no performance benefit. However, there are various MicroStrategy features and techniques that often require multiple queries and therefore can benefit the most from parallel query execution, which include:

- Consolidations and custom groups.
- Level metrics and transformation metrics.
- Accessing multiple data sources using MultiSource Option or database gateway support.
- Accessing data sources that use temporary tables or permanent tables to store intermediate results.
- Accessing data in multiple tables through the use of Data Import.

If your report or Intelligent Cube uses any of the features listed above, it may be a good candidate for using parallel query execution. Additionally, using parallel query execution can be a good option for Intelligent Cubes that are published during off-peak hours when the system is not in heavy use by the reporting community. Using parallel query execution to publish these Intelligent Cubes can speed up the publication process, while not affecting the reporting community for your system.

There are additional scenarios in MicroStrategy that can require multiple queries. To help analyze which reports and Intelligent Cubes may benefit from the use of parallel query execution, you can use the parallel query execution improvement estimate provided in the SQL view of a report or Intelligent Cube. For more information on this estimate and disabling or enabling the inclusion of this estimate, see *Parallel Query Execution, page 162*.

There are some scenarios where parallel query execution cannot be used. These are described below:

- When reports contain user-defined data mart SQL, parallel query execution cannot be used to execute multiple queries in parallel. For information on data mart Pre/Post Statement VLDB properties, including at what levels these VLDB properties can be defined, see *Customizing SQL statements: Pre/Post Statements, page 117*.
- Both MultiSource Option and warehouse partition mapping are used to return results for a report or Intelligent Cube from multiple data sources. While the use of MultiSource Option alone can be a good candidate for parallel query execution, when MultiSource Option is combined with warehouse partition mapping to return results from multiple data sources, parallel query execution cannot be used to execute

multiple queries in parallel. For information on using warehouse partition mapping for a project, see the Project Design Guide.

- Microsoft Access databases support parallel query execution for Intelligent Cubes. However, reports and Intelligent Cubes that require the creation of temporary tables or insertion of values as part of parallel query execution are instead processed sequentially for Access databases.

When to disable parallel query execution

While performing multiple queries in parallel can improve the performance of query execution in MicroStrategy, it will not provide the best performance or results in all scenarios.

Parallel query execution is disabled by default to allow you to first verify that your reports and Intelligent Cubes are executing correctly prior to any parallel query optimization. If you enable parallel query execution and errors are encountered or data is not being returned as expected, disabling parallel query execution can help to troubleshoot the report or Intelligent Cube.

When multiple queries are performed in parallel, the actual processing of the multiple queries is performed in parallel on the database. If a database is required to do too many tasks at the same time this can cause the response time of the database to slow down, and thus degrade the overall performance. You should take into account the databases used to retrieve data and their available resources when deciding whether to enable parallel query execution.

Disabling parallel query execution can be a good option for reports and Intelligent Cubes that are not used often or ones that do not have strict performance requirements. If you can disable parallel query execution for these reports and Intelligent Cubes that do not have a great need for enhanced performance, that can save database resources to handle other potentially more important requests.

Additionally, you can limit the number of queries that can be executed in parallel for a given report or Intelligent Cube. This can allow you to enable parallel query execution, but restrict how much processing can be done in parallel on the database. To define the number of passes of SQL that can be executed in parallel, see *Parallel Query Execution, page 162*.

Levels at which you can set this

Project, report, and template

Parallel Query Execution Improvement Estimate in SQL View

Parallel Query Execution Improvement Estimate in SQL View is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Parallel Query Execution Improvement Estimate in SQL View property determines whether reports and Intelligent Cubes include an estimate in the percent of processing

time that would be saved if parallel query execution was used to run multiple queries in parallel. This VLDB property has the following options:

- **Disable parallel query execution estimate in SQL view**: An estimate on the percent of processing time that would be saved if parallel query execution was used for the report or Intelligent Cube is not displayed in the SQL view of the report. This can simplify the SQL view if you are already using parallel query execution or you are not interested in this estimated improvement.
- **Enable parallel query execution estimate in SQL view**: An estimate on the percent of processing time that would be saved if parallel query execution was used for the report or Intelligent Cube is displayed in the SQL view of the report. This estimate is provided as a percentage of time that could be saved by using parallel query execution.

  To calculate this estimate, the report or Intelligent Cube is analyzed to determine if there are multiple queries. If there are multiple queries, this estimate is calculated by assuming that all applicable queries are run in parallel.

  Be aware that this estimate does not factor in the capabilities of the database you are using, which can have an effect on the performance of parallel query execution since the database is what processes the multiple passes in parallel. Additionally, this estimate assumes that all queries that can be done in parallel are in fact performed in parallel. If parallel query execution is enabled, the number of queries that can be performed in parallel is controlled by the Maximum Parallel SQLs Per Report VLDB property (see *Parallel Query Execution Improvement Estimate in SQL View, page 165*).

Levels at which you can set this

Project only

Rank Method if DB Ranking Not Used

The Rank Method property determines which method to use for ranking calculations. There are three methods for ranking data, and in some cases, this property is ignored. The logic is as follows:

1. If the metric that is being ranked has to be calculated in the Analytical Engine, then the ranking is calculated in the Analytical Engine as well.
2. If the database supports the Rank function, then the ranking is done in the database.
3. If neither of the above criteria is met, then the Rank Method property setting is used.

The most common form of ranking is referred to as Open Database Connectivity (ODBC) ranking. This was the standard method used by MicroStrategy 6.x and earlier. This method makes multiple queries against the warehouse to determine the ranking. ODBC ranking is the default ranking technique, when the database does not support native ranking, because it is more efficient for large datasets.

Analytical Engine ranking generates the result of the ranking operation in the MicroStrategy Analytical Engine and then moves the result set back to the warehouse to perform any further operations and compile the final result set.

Levels at which you can set this

Database instance, report, and template

Remove Aggregation Method

Remove Aggregation Method is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Remove Aggregation Method property determines whether to keep or remove aggregations in SQL queries executed from MicroStrategy. This VLDB property has the following options:

- **Remove aggregation according to key of FROM clause** (default): Aggregations are kept or removed based on the level of data created by joining all the tables included in the query. If the level of the information returned in the query (`SELECT` clause) is the same as the level determined by joining all required tables (`FROM` clause) then any unnecessary aggregations are removed. If these levels are different, then aggregations must be kept to ensure that correct data is returned. Determining whether aggregations are necessary after joining all relevant tables helps to provide valid SQL when the attribute data and the fact data are stored at different levels.

  For example, the report shown in the image below was created in the MicroStrategy Tutorial project.

Month	Customer City	Metrics	Revenue
	Lancaster		$2,080
	Langhorne		$2,242
	Las Cruces		$805
	Las Vegas		$480
	Le Grand		$371
	Leominster		$1,547
	Libby		$1,654
Jan 2007	Lincoln		$1,328
	Lindhurst		$1,041
	Livonia		$354

  To create this report, data must be joined from the tables `LU_MONTH`, `LU_CUST_CITY`, and `CITY_MNTH_SLS`. Since the attribute lookup tables combine to have a level of Customer City and Month, and the `CITY_MNTH_SLS` table has a level of Customer City and Month, normally this VLDB property would have no effect on the SQL. However, for the purposes of this example the `LU_MONTH` table was modified to include an extra attribute named Example, and it is not related to the Month attribute. Because of this additional unrelated attribute, while the report only displays

Month and Customer City, the level of the data is Month, Customer City, and Example. If you use the other option (**Remove aggregation according to key of fact tables**) for this VLDB property, the following SQL is created:

```
select   distinct a11.[MONTH_ID] AS MONTH_ID,
         a13.[MONTH_DESC] AS MONTH_DESC0,
         a11.[CUST_CITY_ID] AS CUST_CITY_ID,
         a12.[CUST_CITY_NAME] AS CUST_CITY_NAME,
         a11.[TOT_DOLLAR_SALES] AS Revenue
from     [CITY_MNTH_SLS] a11,
         [LU_CUST_CITY]  a12,
         [LU_MONTH]      a13
where    a11.[CUST_CITY_ID] = a12.[CUST_CITY_ID] and
         a11.[MONTH_ID] = a13.[MONTH_ID]
```

The SQL statement above uses `DISTINCT` in the `SELECT` clause to return the Month data. However, since there is an additional attribute on the `LU_MONTH` table, the correct SQL to use includes aggregations on the data rather than using `DISTINCT`. Therefore, if you use this **Remove aggregation according to key of FROM clause** option for the VLDB property, the following SQL is created:

```
select   a11.[MONTH_ID] AS MONTH_ID,
         max(a13.[MONTH_DESC]) AS MONTH_DESC0,
         a11.[CUST_CITY_ID] AS CUST_CITY_ID,
         max(a12.[CUST_CITY_NAME]) AS CUST_CITY_NAME,
         sum(a11.[TOT_DOLLAR_SALES]) AS Revenue
from     [CITY_MNTH_SLS] a11,
         [LU_CUST_CITY]  a12,
         [LU_MONTH]      a13
where    a11.[CUST_CITY_ID] = a12.[CUST_CITY_ID] and
         a11.[MONTH_ID] = a13.[MONTH_ID]
group by a11.[MONTH_ID],
         a11.[CUST_CITY_ID]
```

This SQL statement correctly uses aggregation functions and a `GROUP BY` clause to return the attribute data.

- **Remove aggregation according to key of fact tables**: Aggregations are kept or removed prior to determining the level of data created by joining all of the tables required for the query. This option can be used for backward compatibility, which can help to provide the expected data and SQL statements in scenarios that utilize features such as nested aggregation in metrics and custom groups.

Levels at which you can set this

Database instance, report, and template

Remove Group By Option

Remove Group By Option is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Remove Group By Option property determines when `Group By` conditions and aggregations can be omitted in specific SQL generation scenarios, depending on which of the following options you select:

- **Remove aggregation and Group By when Select level is identical to From level** (default): This setting provides the common behavior of omitting `Group By` conditions and aggregations if the level of the `SELECT` clause is identical to the level of the `FROM` clause. For example, a SQL statement that only includes the ID column for the Store attribute in the `SELECT` clause and only includes the lookup table for the Store attribute in the `FROM` clause does not include any `Group By` conditions.
- **Remove aggregation and Group By when Select level contains all attribute(s) in From level**: You can choose to omit `Group By` conditions and aggregations in the unique scenario of including multiple attributes on a report, which are built from columns of the same table in the data warehouse. For example, you have separate attributes for shipping data such as Shipping ID, Shipping Time, and Shipping Location which are mapped to columns in the same table which has the primary key mapped to Shipping ID. By selecting this setting, when these three attributes are included on a report, `Group By` conditions and aggregations for the shipping attributes are omitted.

Levels at which you can set this

Database instance, report, and template

Remove Repeated Tables For Outer Joins

Remove Repeated Tables For Outer Joins is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Remove Repeated Tables For Outer Joins property determines whether an optimization for outer join processing is enabled or disabled. You have the following options:

- **Disable optimization to remove repeated tables in full outer join and left outer join passes**: The optimization for outer join processing is disabled. This can cause outer joins to require additional processing time.
- **Enable optimization to remove repeated tables in full outer join and left outer join passes** (default): The optimization for outer join processing is enabled. This can provide better response time for outer join processing.

  However, if you sort or rank report results and some of the values used for the sort or rank are identical, you may encounter different sort or rank orders depending on whether you disable or enable this optimization. To preserve current sorting or ranking orders on identical values, you may want to disable this optimization.

Levels at which you can set this

Database instance, report, and template

Set Operator Optimization

Set Operator Optimization is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Set Operator Optimization property determines whether to use set operators, such as `EXCEPT` and `INTERSECT`, to combine multiple filter qualifications rather than their equivalent logical operators such as `AND NOT` and `AND`. Set operators can be used to combine two or more of the following types of set qualifications:

- Relationship qualifications
- Metric qualifications when combined with other types of set qualifications with the logical operators `AND`, `NOT`, or `OR`
- Report as filter qualifications when combined with the logical operators `AND`, `NOT`, or `OR`

- Set operators can only be used to combine the filter qualifications listed above if they have the same output level. For example, a relationship qualification with an output level set to Year and Region cannot be combined with another relationship qualification with an output level of Year.
- Metric qualifications and report-as-filter qualifications, when combined with `AND`, render as inner joins by default to avoid a subquery in the final result pass. When Set Operator Optimization is enabled, the inner joins are replaced by subqueries combined using `INTERSECT`.
- Metric qualifications at the same level are combined into one set qualification before being applied to the final result pass. This is more efficient than using a set operator. Consult MicroStrategy Tech Note TN13536 for more details.
- For more information on filters and filter qualifications, see the *Advanced Filters* chapter of the *MicroStrategy Advanced Reporting Guide*.

Along with the restrictions described above, SQL set operators also depend on the subquery type and the database platform. For more information on sub query type, see *Set Operator Optimization, page 170*. Set Operator Optimization can be used with the following sub query types:

- WHERE COL1 IN (SELECT s1.COL1...) falling back to EXISTS (SELECT * ...) for multiple columns IN
- WHERE (COL1, COL2...) IN (SELECT s1.COL1, s1.COL2...)
- WHERE COL1 IN (SELECT s1.COL1...) falling back to EXISTS (SELECT col1, col2 ...) for multiple columns IN
- Use Temporary Table, falling back to IN (SELECT COL) for correlated sub query

If either of the two sub query types that use fallback actions perform a fallback, Set Operator Optimization is not applied.

The following database platforms support SQL set operators:

Database	Intersect	Intersect ALL	Except	Except ALL	Union	Union ALL
ANSI 92	Yes	Yes	Yes	Yes	Yes	Yes
DB2 UDB	Yes	Yes	Yes	Yes	Yes	Yes
Informix	No	No	No	No	Yes	Yes
Oracle	Yes	No	Yes (Minus)	No	Yes	Yes
RedBrick	Yes	Yes	Yes	Yes	Yes	Yes
SQL Server	Yes (2005 and later)	No	Yes (2005 and later)	No	Yes	Yes
Tandem	No	No	No	No	No	No
Teradata	Yes	Yes	Yes	Yes	Yes	Yes

If you enable Set Operator Optimization for a database platform that does not support operators such as `EXCEPT` and `INTERSECT`, the Set Operator Optimization property is ignored.

The Set Operator Optimization property provides you with the following options:

- **Disable Set Operator Optimization** (default): Operators such as `IN` and `AND NOT` are used in SQL sub queries with multiple filter qualifications.
- **Enable Set Operator Optimization (if database support and [Sub Query Type])**: This setting can improve performance by using SQL set operators such as `EXCEPT`, `INTERSECT`, and `MINUS` in SQL sub queries to combine multiple filter qualifications that have the same output level. All of the dependencies described above must be met for SQL set operators to be used. If you enable SQL set operators for a database platform that does not support them, this setting is ignored and filters are combined in the standard way with operators such as `IN` and `AND NOT`.

For a further discussion on the Set Operator Optimization VLDB property, refer to MicroStrategy Tech Note TN13530.

Levels at which you can set this

Database instance, report, and template

SQL Global Optimization

The SQL Global Optimization property provides access to level options you can use to determine whether and how SQL queries are optimized.

In some cases, the SQL Engine generates duplicate or redundant passes, generates SQL passes that can be combined into one pass, or creates unnecessary temporary tables. Such SQL queries can have an adverse effect on performance.

The default option for this VLDB property has changed in 9.0.0. For information on this change, see *SQL Global Optimization, page 172*.

You can set the following SQL Global Optimization options to determine the extent to which SQL queries are optimized:

- **Level 0: No optimization**: SQL queries are not optimized.
- **Level 1: Remove Unused and Duplicate Passes**: Redundant, identical, and equivalent SQL passes are removed from queries during SQL generation.
- **Level 2: Level 1 + Merge Passes with different SELECT**: Level 1 optimization takes place as described above, and SQL passes from different `SELECT` statements are consolidated when it is appropriate to do so.
- **Level 3: Level 2 + Merge Passes, which only hit DB Tables, with different WHERE**: Level 2 optimization takes place as described above, and SQL passes which access database tables with different WHERE clauses are consolidated when it is appropriate to do so.
- **Level 4: Level 2 + Merge All Passes with Different WHERE**: This is the default level. Level 2 optimization takes place as described above, and all SQL passes with different WHERE clauses are consolidated when it is appropriate to do so. While Level 3 only consolidates SQL statements that access database tables, this option also considers SQL statements that access temporary tables, derived tables, and common table expressions.
- **Level 5: Level 2 + Merge All Passes, which hit the same warehouse fact tables**: Level 2 optimization takes place as described above, and when multiple passes hit the same fact table, a compiled table is created from the lookup tables of the multiple passes. This compiled table hits the warehouse fact table only once.

Additionally, if you use either Level 3 or Level 4 SQL Global Optimization, SQL passes can also be combined for the SQL that is generated for separate custom group elements.

The SQL optimization available with Level 3 or Level 4 can be applied for SQL passes that use the functions Plus (+), Minus (–), Times (*), Divide (/), Unary minus (U-), Sum, Count, Avg (average), Min, and Max. To ensure that valid SQL is returned, if the SQL passes that are generated use any other functions, the SQL passes are not combined.

Example: Redundant SQL pass

This example demonstrates how some SQL passes are redundant and therefore removed when the Level 1 or Level 2 SQL Global Optimization option is selected.

Suppose the following appear on the report template:

- Year attribute

- Region attribute
- Sum(Profit) {~+, Category%} metric (calculates profit for each Category, ignoring any filtering on Category)

The report generates the following SQL:

- **SQL Pass 1**: Retrieves the set of categories that satisfy the metric qualification

```
SELECT a11.CATEGORY_ID  CATEGORY_ID
into #ZZTRH02012JMQ000
FROM YR_CATEGORY_SLS a11
GROUP BY a11.CATEGORY_ID
HAVING sum(a11.TOT_DOLLAR_SALES) > 1000000.0
```

- **SQL Pass 2**: Final pass that selects the related report data, but does not use the results of the first SQL pass

```
SELECT a13.YEAR_ID  YEAR_ID,
a12.REGION_ID  REGION_ID,
max(a14.REGION_NAME)  REGION_NAME,
sum((a11.TOT_DOLLAR_SALES - a11.TOT_COST))
WJXBFS1
FROM DAY_CTR_SLS a11
join LU_CALL_CTR a12
on (a11.CALL_CTR_ID = a12.CALL_CTR_ID)
join LU_DAY a13
on (a11.DAY_DATE = a13.DAY_DATE)
join LU_REGION a14
on (a12.REGION_ID = a14.REGION_ID)
GROUP BY a13.YEAR_ID,a12.REGION_ID
```

SQL Pass 1 is redundant because it creates and populates a temporary table, `#ZZTRH02012JMQ000`, that is not accessed again and is unnecessary to generating the intended SQL result.

If you select either the **Level 1: Remove Unused and Duplicate Passes** or **Level 2: Level 1 + Merge Passes with different SELECT** option, only one SQL pass—the second SQL pass described above—is generated because it is sufficient to satisfy the query on its own. By selecting either option, you reduce the number of SQL passes from two to one, which can potentially decrease query time.

Example: Combinable SQL passes

Sometimes, two or more passes contain SQL that can be consolidated into a single SQL pass, as shown in the example below. In such cases, you can select the **Level 2: Level 1 + Merge Passes with different SELECT** option to combine multiple passes from different SELECT statements.

Suppose the following appear on the report template:

- Region attribute
- Metric 1 = Sum(Revenue) {Region+} (calculates the total revenue for each region)
- Metric 2 = Count<FactID=Revenue>(Call Center) {Region+} (calculates the number of call centers for each region)

- Metric 3 = Metric 1/Metric 2 (Average Revenue = Total Revenue/Number of Call Centers)

The report generates the following SQL:

- **SQL Pass 1**: Calculates Metric 1 = Sum(Revenue) {Region+}

```
        SELECT a12.[REGION_ID] AS REGION_ID,
sum(a11.[TOT_DOLLAR_SALES]) AS WJXBFS1
        into [ZZTI10200U2MD000]
        FROM [CITY_CTR_SLS] a11,
        [LU_CALL_CTR] a12
        WHERE a11.[CALL_CTR_ID] = a12.[CALL_CTR_ID]
        GROUP BY a12.[REGION_ID]
```

- **SQL Pass 2**: Calculates Metric 2 = Count<FactID=Revenue>(Call Center) {Region+}

```
        SELECT a12.[REGION_ID] AS REGION_ID,
count(a11.[CALL_CTR_ID]) AS WJXBFS1
        into [ZZTI10200U2MD001]
        FROM [CITY_CTR_SLS] a11,
        [LU_CALL_CTR] a12
        WHERE a11.[CALL_CTR_ID] = a12.[CALL_CTR_ID]
        GROUP BY a12.[REGION_ID]
```

- **SQL Pass 3**: Final pass that calculates Metric 3 = Metric 1/Metric 2 and displays the result

```
        SELECT pa11.[REGION_ID] AS REGION_ID,
        a13.[REGION_NAME] AS REGION_NAME,
        pa11.[WJXBFS1] AS WJXBFS1,
        IIF(ISNULL((pa11.[WJXBFS1] / IIF(pa12.[WJXBFS1]
        = 0, NULL,
        pa12.[WJXBFS1]))), 0,
        (pa11.[WJXBFS1] / IIF(pa12.[WJXBFS1] = 0,
        NULL,pa12.[WJXBFS1]))) AS WJXBFS2
        FROM [ZZTI10200U2MD000] pa11,
        [ZZTI10200U2MD001] pa12,
        [LU_REGION] a13
        WHERE pa11.[REGION_ID] = pa12.[REGION_ID] and
        pa11.[REGION_ID] = a13.[REGION_ID]
```

Because SQL passes 1 and 2 contain almost exactly the same code, they can be consolidated into one SQL pass. Notice the italicized SQL in Pass 1 and Pass 2. These are the only unique characteristics of each pass; therefore, Pass 1 and 2 can be combined into just one pass. Pass 3 remains as it is.

You can achieve this type of optimization by selecting the **Level 2: Level 1 + Merge Passes with different SELECT** option. The SQL that results from this level of SQL optimization is as follows:

Pass 1:

```
SELECT a12.[REGION_ID] AS REGION_ID,
count(a11.[CALL_CTR_ID]) AS WJXBFS1
sum(a11.[TOT_DOLLAR_SALES]) AS WJXBFS1
into [ZZTI10200U2MD001]
FROM [CITY_CTR_SLS] a11,
[LU_CALL_CTR] a12
WHERE a11.[CALL_CTR_ID] = a12.[CALL_CTR_ID]
GROUP BY a12.[REGION_ID]
```

Pass 2:

```
SELECT pa11.[REGION_ID] AS REGION_ID,
a13.[REGION_NAME] AS REGION_NAME,
pa11.[WJXBFS1] AS WJXBFS1,
IIF(ISNULL((pa11.[WJXBFS1] / IIF(pa12.[WJXBFS1] = 0, NULL,
pa12.[WJXBFS1]))), 0,
(pa11.[WJXBFS1] / IIF(pa12.[WJXBFS1] = 0, NULL,
pa12.[WJXBFS1]))) AS WJXBFS2
FROM [ZZTI10200U2MD000] pa11,
[ZZTI10200U2MD001] pa12,
[LU_REGION] a13
WHERE pa11.[REGION_ID] = pa12.[REGION_ID] and
pa11.[REGION_ID] = a13.[REGION_ID]
```

For a further discussion on the SQL Global Optimization VLDB property, refer to MicroStrategy Tech Note TN13505.

Example: Combinable SQL passes, with different WHERE clauses

Sometimes, two or more passes contain SQL with different where clauses that can be consolidated into a single SQL pass, as shown in the example below. In such cases, you can select the **Level 3: Level 2 + Merge Passes, which only hit DB Tables, with different WHERE** option or the **Level 4: Level 2 + Merge All Passes with Different WHERE** option to combine multiple passes with different WHERE clauses.

Suppose the following appear on the report template:

- Quarter attribute
- Metric 1 = Web Sales (Calculates sales for the web call center)
- Metric 2 = Non-Web Sales (Calculates sales for all non-web call centers)

The report generates the following SQL

Pass 1:

```
create table ZZMD00 (
        QUARTER_ID      SHORT,
        WJXBFS1 DOUBLE)
```

Pass 2:

```
insert into ZZMD00
select  a12.[QUARTER_ID] AS QUARTER_ID,
        sum(a11.[TOT_DOLLAR_SALES]) AS WJXBFS1
from     [DAY_CTR_SLS]  a11,
        [LU_DAY]        a12
where   a11.[DAY_DATE] = a12.[DAY_DATE]
 and    a11.[CALL_CTR_ID] in (18)
group by        a12.[QUARTER_ID]
```

Pass 3:

```
create table ZZMD01 (
        QUARTER_ID      SHORT,
        WJXBFS1 DOUBLE)
```

Pass 4:

```
insert into ZZMD01
```

```
select  a12.[QUARTER_ID] AS QUARTER_ID,
        sum(a11.[TOT_DOLLAR_SALES]) AS WJXBFS1
from     [DAY_CTR_SLS]  a11,
        [LU_DAY]       a12
where    a11.[DAY_DATE] = a12.[DAY_DATE]
 and     a11.[CALL_CTR_ID] not in (18)
group by        a12.[QUARTER_ID]
```

Pass 5:

```
select  pa11.[QUARTER_ID] AS QUARTER_ID,
        a13.[QUARTER_DESC] AS QUARTER_DESC0,
        pa11.[WJXBFS1] AS WJXBFS1,
        pa12.[WJXBFS1] AS WJXBFS2
from    [ZZMD00]        pa11,
        [ZZMD01]        pa12,
        [LU_QUARTER]    a13
where    pa11.[QUARTER_ID] = pa12.[QUARTER_ID] and
        pa11.[QUARTER_ID] = a13.[QUARTER_ID]
```

Pass 2 calculates the Web Sales and Pass 4 calculates all non-Web Sales. Because SQL passes 2 and 4 contain almost exactly the same SQL, they can be consolidated into one SQL pass. Notice the highlighted SQL in Pass 2 and Pass 4. These are the only unique characteristics of each pass; therefore, Pass 2 and 4 can be combined into just one pass.

You can achieve this type of optimization by selecting the **Level 3: Level 2 + Merge Passes, which only hit DB Tables, with different WHERE** option or the **Level 4: Level 2 + Merge All Passes with Different WHERE** option. The SQL that results from this level of SQL optimization is as follows:

Pass 1:

```
create table ZZT6C00009GMD000 (
        QUARTER_ID      SHORT,
        WJXBFS1 DOUBLE,
        GODWFLAG1_1     LONG,
        WJXBFS2 DOUBLE,
        GODWFLAG2_1     LONG)
```

Pass 2:

```
insert into ZZT6C00009GMD000
select  a12.[QUARTER_ID] AS QUARTER_ID,
        sum(iif(a11.[CALL_CTR_ID] in (18),
               a11.[TOT_DOLLAR_SALES], NULL))
               AS WJXBFS1,
        max(iif(a11.[CALL_CTR_ID] in (18), 1, 0))
               AS GODWFLAG1_1,
        sum(iif(a11.[CALL_CTR_ID] not in (18),
               a11.[TOT_DOLLAR_SALES], NULL))
               AS WJXBFS2,
        max(iif(a11.[CALL_CTR_ID] not in (18), 1, 0))
               AS GODWFLAG2_1
from    [DAY_CTR_SLS]   a11,
        [LU_DAY]        a12
where    a11.[DAY_DATE] = a12.[DAY_DATE]
 and    (a11.[CALL_CTR_ID] in (18)
 or a11.[CALL_CTR_ID] not in (18))
group by        a12.[QUARTER_ID]
```

Pass 3:

```
select  pa12.[QUARTER_ID] AS QUARTER_ID,
        a13.[QUARTER_DESC] AS QUARTER_DESC0,
```

```
        pa12.[WJXBFS1] AS WJXBFS1,
        pa12.[WJXBFS2] AS WJXBFS2
from    [ZZT6C00009GMD000]      pa12,
        [LU_QUARTER]    a13
where    pa12.[QUARTER_ID] = a13.[QUARTER_ID]
 and    (pa12.[GODWFLAG1_1] = 1
 and pa12.[GODWFLAG2_1] = 1)
```

Upgrading from pre-9.0.x versions of MicroStrategy

The default option for the SQL Global Optimization VLDB property changed in MicroStrategy 9.0.0. In pre-9.0.x versions of MicroStrategy, the default option for this VLDB property was Level 2: Level 1 + Merge Passes with different SELECT. Starting with MicroStrategy 9.0.0, the default option for this VLDB property is Level 4: Level 2 + Merge All Passes with Different WHERE.

When projects are upgraded to 9.0.x, if you have defined this VLDB property to use the default setting, this new default is applied. This change improves performance for the majority of reporting scenarios. However, the new default can cause certain reports to become unresponsive or fail with time-out errors. For example, reports that contain custom groups or a large number of conditional metrics may encounter performance issues with this new default.

You can use Integrity Manager to determine any changes in performance that your reports may encounter due to upgrading your MicroStrategy projects. This allows you to determine which reports may encounter performance issues due to this VLDB property modification.

To resolve this issue for a report, after completing an upgrade, modify the SQL Global Optimization VLDB property for the report to use the option Level 2: Level 1 + Merge Passes with different SELECT.

Levels at which you can set this

Database instance, report, and template

Sub Query Type

Sub Query Type is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Sub Query Type property tells the Analytical Engine what type of syntax to use when generating a subquery. A subquery is a secondary `SELECT` statement in the `WHERE` clause of the primary SQL statement.

The Sub Query Type property is database specific, due to the fact that different databases have different syntax support for subqueries. Some databases can have improved query building and performance depending on the subquery type used. For example, it is more efficient to use a subquery that only selects the needed columns rather than selecting every column. Subqueries can also be more efficient by using the `IN` clause rather than using the `EXISTS` function.

The most optimal option depends on your database capabilities. In general the default setting is **WHERE COL1 IN (SELECT s1.COL1...) falling back to EXISTS (SELECT *...) for multiple columns IN**. However, the default setting is based on the most optimal setting for your database type. See the table below for database platform exceptions to the default setting. To review example SQL syntax for each VLDB setting for Sub Query Type, see *Examples, page 178*.

Levels at which you can set this

Database instance, report, and template

Database exceptions to the default setting

Database	Default
DB2 UDB	Use Temporary Table, falling back to EXISTS (SELECT *...) for correlated subquery
DB2 UDB for OS/390	Where Exists (Select *...)
Microsoft Access 2000/2002/2003	Use Temporary Table, falling back to EXISTS (SELECT *...) for correlated subquery
Microsoft Excel 2000/2003	Use Temporary Table, falling back to EXISTS (SELECT *...) for correlated subquery
Netezza	Where (col1, col2...) in (Select s1.col1, s1.col2...)
Oracle	Where (col1, col2...) in (Select s1.col1, s1.col2...)
PostgreSQL	Where (col1, col2...) in (Select s1.col1, s1.col2...)
RedBrick	Where col1 in (Select s1.col1...) falling back to Exists (Select col1, col2...) for multiple column in
Teradata	Use Temporary Table, falling back to in (Select col) for correlated subquery

Notice that some options have a fallback action. In some scenarios, the selected option does not work, so the SQL Engine must fall back to an approach that always works. The typical scenario for falling back is when multiple columns are needed in the `IN` list, but the database does not support it and the correlated subqueries.

For a further discussion of the Sub Query Type VLDB property, refer to MicroStrategy Tech Note TN13870.

Examples

WHERE EXISTS (Select *...)

```
select a31.ITEM_NBR ITEM_NBR,
        a31.CLASS_NBR CLASS_NBR,
        sum(a31.REG_SLS_DLR) REG_SLS_DLR
from REGION_ITEM a31
where (exists (select *
```

```
        from REGION_ITEM r21,
        LOOKUP_DAY r22
        where r21.CUR_TRN_DT = r22.CUR_TRN_DT
        and r22.SEASON_ID in (199501)
        and r21.ITEM_NBR = a31.ITEM_NBR
        and r21.CLASS_NBR = a31.CLASS_NBR))
group by a31.ITEM_NBR,
        a31.CLASS_NBR
```

WHERE EXISTS (SELECT col1, col2...)

```
select a31.ITEM_NBR ITEM_NBR,
        a31.CLASS_NBR CLASS_NBR,
        sum(a31.REG_SLS_DLR) REG_SLS_DLR
from REGION_ITEM a31
where (exists (select a31.ITEM_NBR ITEM_NBR,
        a31.CLASS_NBR CLASS_NBR
        from REGION_ITEM r21,
        LOOKUP_DAY r22
        where r21.CUR_TRN_DT = r22.CUR_TRN_DT
        and r22.SEASON_ID in (199501)
        and r21.ITEM_NBR = a31.ITEM_NBR
        and r21.CLASS_NBR = a31.CLASS_NBR))
group by a31.ITEM_NBR,
        a31.CLASS_NBR
```

WHERE COL1 IN (SELECT s1.COL1...) falling back to EXISTS (SELECT * ...) for multiple columns IN

```
select a31.ITEM_NBR ITEM_NBR,
        sum(a31.REG_SLS_DLR) REG_SLS_DLR
from REGION_ITEM a31
where ((a31.ITEM_NBR)
        in (select r21.ITEM_NBR ITEM_NBR,
        from REGION_ITEM r21,
        LOOKUP_DAY r22
        where r21.CUR_TRN_DT = r22.CUR_TRN_DT
        and r22.SEASON_ID in (199501)))
group by a31.ITEM_NBR
```

WHERE (COL1, COL2...) IN (SELECT s1.COL1, s1.COL2...)

```
select a31.ITEM_NBR ITEM_NBR,
        a31.CLASS_NBR CLASS_NBR,
        sum(a31.REG_SLS_DLR) REG_SLS_DLR
from REGION_ITEM a31
where ((a31.ITEM_NBR,
        a31.CLASS_NBR)
        in (select r21.ITEM_NBR ITEM_NBR,
        r21.CLASS_NBR CLASS_NBR
        from REGION_ITEM r21,
        LOOKUP_DAY r22
        where r21.CUR_TRN_DT = r22.CUR_TRN_DT
        and r22.SEASON_ID in (199501)))
group by a31.ITEM_NBR,
        a31.CLASS_NBR
```

Use Temporary Table, falling back to EXISTS (SELECT *...) for correlated subquery (default)

```
create table TEMP1 as
select r21.ITEM_NBR ITEM_NBR,
from REGION_ITEM r21,
        LOOKUP_DAY r22
where r21.CUR_TRN_DT = r22.CUR_TRN_DT
```

```
        and r22.SEASON_ID in 199501
select a31.ITEM_NBR ITEM_NBR,
        sum(a31.REG_SLS_DLR) REG_SLS_DLR
from REGION_ITEM a31
        join TEMP1 a32
        on a31.ITEM_NBR = a32.ITEM_NBR
group by a31.ITEM_NBR
```

WHERE COL1 IN (SELECT s1.COL1...) falling back to EXISTS (SELECT col1, col2 ...) for multiple columns IN

```
select a31.ITEM_NBR ITEM_NBR,
        sum(a31.REG_SLS_DLR) REG_SLS_DLR
from REGION_ITEM a31
where ((a31.ITEM_NBR)
        in (select r21.ITEM_NBR ITEM_NBR,
        from REGION_ITEM r21,
        LOOKUP_DAY r22
        where r21.CUR_TRN_DT = r22.CUR_TRN_DT
        and r22.SEASON_ID in (199501)))
group by a31.ITEM_NBR
```

Use Temporary Table, falling back to IN (SELECT COL) for correlated subquery

```
create table TEMP1 as
select r21.ITEM_NBR ITEM_NBR,
from REGION_ITEM r21,
        LOOKUP_DAY r22
where r21.CUR_TRN_DT = r22.CUR_TRN_DT
        and r22.SEASON_ID in 199501
select a31.ITEM_NBR ITEM_NBR,
        sum(a31.REG_SLS_DLR) REG_SLS_DLR
from REGION_ITEM a31
        join TEMP1 a32
        on a31.ITEM_NBR = a32.ITEM_NBR
group by a31.ITEM_NBR
```

Transformation Formula Optimization

Transformation Formula Optimization is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Transformation Formula Optimization VLDB property allows you to improve the performance of expression-based transformations. Performance can be improved for reports that include expression-based transformations and meet the following requirements:

- No attributes on the report grid or the Report Objects of the report are related to the transformation's member attribute. For example, if a transformation is defined on the attribute Year of the Time hierarchy, no attributes in the Time hierarchy can be included on the report grid or Report Objects.
- The filter of the report does contain attributes that are related to the transformation's member attribute. For example, if a transformation is defined on the attribute Year of the Time hierarchy, a filter on another attribute in the Time hierarchy is included on the report.

For information on expression-based transformations and how to create them, see the Project Design Guide.

If your report includes an expression-based transformation This VLDB property has the following options:

- **Always join with transformation table to perform transformation**: A join with the transformation table is used to perform the transformation. This option supports backwards compatibility and also serves as a fallback if optimization cannot be applied for the transformation.
- **Use transformation formula instead of join with transformation table when possible** (default): If the transformation is an expression-based transformation and the report meets the requirements listed above, the expression is used rather than using a join with the transformation table.

  This can improve performance of expression-based transformations by eliminating the requirement to join with the transformation table. If the transformation is included on a report that cannot support this optimization, then a join with the transformation table is automatically used to support the transformation. An example of this optimization is shown below.

Levels at which you can set this

Database instance, report, and template

Example

The SQL statements shown below display a SQL statement before (Statement 1) and after (Statement 2) applying the transformation optimization.

Statement 1

```
select a14.CATEGORY_ID CATEGORY_ID,
max(a15.CATEGORY_DESC) CATEGORY_DESC,
sum((a11.QTY_SOLD * (a11.UNIT_PRICE - a11.DISCOUNT)))
WJXBFS1
from ORDER_DETAIL a11
        join   LU_DAY a12
        on (a11.ORDER_DATE = a12.DAY_DATE - 1 YEAR)
        join LU_ITEM a13
        on (a11.ITEM_ID = a13.ITEM_ID)
        join LU_SUBCATEG a14
        on (a13.SUBCAT_ID = a14.SUBCAT_ID)
        join LU_CATEGORY a15
        on (a14.CATEGORY_ID = a15.CATEGORY_ID)
where a12.DAY_DATE = '08/31/2001'
group by a14.CATEGORY_ID
```

Statement 2

```
select a14.CATEGORY_ID CATEGORY_ID,
   max(a15.CATEGORY_DESC) CATEGORY_DESC,
   sum((a11.QTY_SOLD * (a11.UNIT_PRICE - a11.DISCOUNT)))
WJXBFS1
from ORDER_DETAIL a11
        join LU_ITEM a13
        on (a11.ITEM_ID = a13.ITEM_ID)
        join LU_SUBCATEG a14
```

```
        on (a13.SUBCAT_ID = a14.SUBCAT_ID)
        join LU_CATEGORY a15
        on (a14.CATEGORY_ID = a15.CATEGORY_ID)
where a11.ORDER_DATE = DATE('08/31/2001') - 1 YEAR
group by a14.CATEGORY_ID
```

Unrelated Filter Options

Unrelated Filter Options is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

MicroStrategy contains the logic to ignore filter qualifications that are not related to the template attributes, to avoid unnecessary Cartesian joins. However, in some cases a relationship is created that should not be ignored. The Unrelated Filter Options property determines whether to remove or keep unrelated filter qualifications that are included in the report's filter or through the use of joint element lists. This VLDB property has the following options:

If filter qualifications are included as part of a report as filter, all filter qualifications are kept on the report regardless of whether they are related or unrelated to the attributes on the report. For information on using the report as filter functionality, see the *Advanced Reporting Guide*.

- **Remove unrelated filter** (default): Any filter qualification with attributes that are unrelated to any of the attributes on the report are removed. An example of how this option can modify a report, in comparison to the **Keep unrelated filter and put condition from unrelated attributes in one subquery group** option, is provided below.
- **Keep unrelated filter**: This option is for backward compatibility. You should switch to using the **Keep unrelated filter and put condition from unrelated attributes in one subquery group** option described below.
- **Keep unrelated filter and put condition from unrelated attributes in one subquery group**: Filter qualifications that include attributes that are unrelated to any of the attributes on the report are kept on the report in certain scenarios. This means that the filtering is applied to the report. However, not all unrelated filter qualifications are kept on a report if you select this option.

  For example, you have report with a filter on the Country attribute, and the Year attribute is on the report template. This example assumes that no relationship between Country and Year is defined in the schema. In this case, the filter is removed regardless of this VLDB property setting. This is because the filter qualification does not include any attributes that could be related to the attributes on the report.

  This setting does keep filter qualifications in certain scenarios. For example, you have a report that is defined as follows:

  - Report filters:
    - Filter 1= (Country, Quarter) in {(England, 2008 Q3), (France, 2008 Q1)}
  - Report template: Includes the Year attribute

Filter 1 described above could be from a joint element list or a combination of report filter qualifications. Since this filter qualification includes the Quarter attribute, which is related to the Year attribute, selecting this option includes the filtering in the reports. The SQL generated with each setting is as follows:

- Removed unrelated filter: The filter qualifications on Country are removed from the report and the report SQL, as shown below:

```
select  distinct a11.[YEAR_ID] AS YEAR_ID
from    [LU_QUARTER] a11
where   (a11.[QUARTER_ID] = 20083
or a11.[QUARTER_ID] = 20081)
```

- Keep unrelated filter and put condition from unrelated attributes in one subquery group: The filter qualifications on Country are included on the report and in the report SQL, as shown below:

```
create table ZZSQ00 (
        QUARTER_ID      SHORT,
        GODWFLAG1_1     LONG,
        GODWFLAG2_1     LONG)
insert into ZZSQ00
select distinct s22.[QUARTER_ID] AS QUARTER_ID,
iif((s21.[COUNTRY_ID] = 3 and s22.[QUARTER_ID] =
20083), 1, 0) AS GODWFLAG1_1,
iif((s21.[COUNTRY_ID] = 4 and s22.[QUARTER_ID] =
20081), 1, 0) AS GODWFLAG2_1
from [LU_COUNTRY]       s21,
[LU_QUARTER]    s22
where   ((s21.[COUNTRY_ID] = 3
and s22.[QUARTER_ID] = 20083)
or (s21.[COUNTRY_ID] = 4
and s22.[QUARTER_ID] = 20081))
select  distinct a13.[YEAR_ID] AS YEAR_ID
from    [ZZSQ00]        pa11,
[ZZSQ00]        pa12,
[LU_QUARTER]    a13
where   pa11.[QUARTER_ID] = pa12.[QUARTER_ID] and
pa11.[QUARTER_ID] = a13.[QUARTER_ID]
and (pa11.[GODWFLAG1_1] = 1
and pa12.[GODWFLAG2_1] = 1)
```

Levels at which you can set this

Database instance, report, and template

Unrelated Filter Options for Nested Metrics

Unrelated Filter Options for Nested Metrics is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Unrelated Filter Options property determines whether to remove or keep unrelated filter qualifications when using nested metrics. Nested metrics, or nested aggregation metrics, are a type of simple metric, where one aggregation function is enclosed inside another. For additional information on nested metrics, see the Advanced Reporting Guide.

To explain how this VLDB property determines whether to keep or remove unrelated filter qualifications when using nested metrics, consider the following example:

- The following example was created in the MicroStrategy Tutorial project, with its data stored in a Microsoft Access database.
- A report is created that includes the following:
  - The Category attribute on the rows of the report.
  - A metric on the columns of the report. THe metric is defined as `Sum (ApplySimple("IIf(#0 = 1, #1, 0)", Region@ID, Sum(Revenue) {~+})) {~}`. This metric returns revenue data for the Northeast region (`Region@ID =1`) or a zero value.
  - A report filter that is defined as `Category In List (Books)`. This report filter returns data only for the Books category.

For the example explained above, the metric includes the Region attribute (through the use of `Region@ID`) and the report filter includes the Category attribute. Since the Category attribute is unrelated to the Region attribute, it is considered unrelated to the nested metric's inclusion of the Region attribute.

This VLDB property has the following options:

- **Use the 8.1.x behavior** (default): Select this option to use the behavior in MicroStrategy 8.1.x. In the example described above, this returns the following SQL statement, which has been abbreviated for clarity:

```
insert into ZZTTM6REM4ZMD000
select a11.[CATEGORY_ID] AS CATEGORY_ID,
sum(a11.[TOT_DOLLAR_SALES]) AS WJXBFS1
from [YR_CATEGORY_SLS] a11
where a11.[CATEGORY_ID] in (1)
group by a11.[CATEGORY_ID]
select    pa11.[CATEGORY_ID] AS CATEGORY_ID,
max(a13.[CATEGORY_DESC]) AS CATEGORY_DESC0,
sum(IIf(a12.[REGION_ID] = 1, pa11.[WJXBFS1], 0))
AS WJXBFS1
from [ZZTTM6REM4ZMD000] pa11,
[LU_REGION] a12,
[LU_CATEGORY] a13
where pa11.[CATEGORY_ID] = a13.[CATEGORY_ID]
group by pa11.[CATEGORY_ID]
```

  While the unrelated filter qualification is kept in the first pass of SQL, it is removed from the second pass of SQL. This means that the filtering on Category is applied to the inner aggregation that returns a summation of revenue for the Northeast region only. However, the filtering on category is not used in the final summation.

  This option can be beneficial for the processing of security filters, which can create additional unrelated filter qualifications on a report based on a user's security filter constraints. Selecting this option can remove some of these unrelated filter qualifications caused by a user's security filter.

- **Use the 9.0.x behavior**: Select this option to use the behavior in MicroStrategy 9.0.x. In the example described above, this returns the following SQL statement, which has been abbreviated for clarity:

```
insert into ZZTTM6REM4ZMD000
select a11.[CATEGORY_ID] AS CATEGORY_ID,
```

```
        sum(a11.[TOT_DOLLAR_SALES]) AS WJXBFS1
        from [YR_CATEGORY_SLS] a11
        where a11.[CATEGORY_ID] in (1)
        group by a11.[CATEGORY_ID]
        select    pa11.[CATEGORY_ID] AS CATEGORY_ID,
        max(a13.[CATEGORY_DESC]) AS CATEGORY_DESC0,
        sum(IIf(a12.[REGION_ID] = 1, pa11.[WJXBFS1], 0))
        AS WJXBFS1
        from [ZZTTM6REM4ZMD000] pa11,
        [LU_REGION] a12,
        [LU_CATEGORY] a13
        where pa11.[CATEGORY_ID] = a13.[CATEGORY_ID]
and pa11.[CATEGORY_ID] in (1)
        group by pa11.[CATEGORY_ID]
```

By using the 9.0.x behavior, the unrelated filter qualification is kept in both SQL passes. This means that the filtering on category is applied to the inner aggregation that returns a summation of revenue for the Northeast region only. The filtering on category is also used in the final summation.

Levels at which you can set this

Database instance, report, and template

WHERE Clause Driving Table

The Where Clause Driving Table property tells the Analytical Engine what type of column is preferred in a qualification of a WHERE clause when generating SQL. One SQL pass usually joins fact tables and lookup tables on certain ID columns. When a qualification is defined on such a column, the Analytical Engine can use the column in either the fact table or the lookup table. In certain databases, like Teradata and RedBrick, a qualification on the lookup table can achieve better performance. By setting the Where Clause Driving Table property to Use Lookup Table, the Analytical Engine always tries to pick the column from the lookup table.

If Use lookup table is selected, but there is no lookup table in the FROM clause for the column being qualified on, the Analytical Engine does not add the lookup table to the FROM clause. To make sure that a qualification is done on a lookup table column, the DSS Star Join property should be set to use Partial star join.

Levels at which you can set this

Database instance, report, and template

Selecting and inserting data with SQL: Select/Insert

The following table summarizes the Select/Insert VLDB properties. Additional details about each property, including examples where necessary, are provided in the sections following the table.

Property	Description	Possible Values	Default Value
Attribute Selection and Form Selection Option for Intermediate Passes	Allows you to choose whether to select attribute forms that are on the template in the intermediate pass (if available).	• Select ID form only • Select ID and other forms if they are on template and available in existing join tree	Select ID form only
Attribute Form Selection Option for Intermediate Pass	Allows you to choose whether to select additional attributes (usually parent attributes) needed on the template as the join tree and their child attributes have already been selected in the **Attribute Form Selection option for Intermediate Pass**.	• (Default) Select only the attributes needed • Select other attributes in current join tree if they are on template and their child attributes have already been selected.	Select only the attributes needed
Selecting and inserting data with SQL: Select/Insert	Determines whether multiple insert statements are issued in the ODBC call, and if together, the string to connect the multiple insert statements.	User-defined	NULL
Constant Column Mode	Allows you to choose whether to use a GROUP BY and how the GROUP BY should be constructed when working with a column that is a constant.	• Pure select, no group by • Use max, no group by • Group by column (expression) • Group by alias • Group by position	Pure select, no group by
Custom Group Interaction With the Report Filter	Allows you define how a report filter interacts with a custom group.	• No interaction - static custom group • Apply report filter to custom group • Apply report filter to custom group, but ignore related elements from the report filter	No interaction - static custom group
Data Retrieval Mode	Determines whether data is retrieved using third-party, native APIs.	• Only ODBC • Allow Native API	Only ODBC
Data Retrieval Parameters	Defines the parameters used to retrieve data using third-party, native APIs.	User-defined	NULL

Property	Description	Possible Values	Default Value
Data Retrieval Mode	Allows you to determine the order in which datamart columns are created.	• Columns created in order based on attribute weight • Columns created in order in which they appear on the template	Columns created in order based on attribute weight
Date Format	Sets the format for date in engine-generated SQL.	User-defined	yyyy-mm-dd
Date Pattern	Lets you define the syntax pattern for Date data.	User-defined	NULL
Decimal Separator	Used to change the decimal separator in SQL statements from a decimal point to a comma, for international database users.	• Use "." as decimal separator (ANSI standard) • Use "," as decimal separator	Use "." as decimal separator (ANSI standard)
Default Attribute Weight	Use this to determine how attributes are treated, for those attributes that are not in the attribute weights list.	• Lowest weight • Highest weight	Highest weight
Disable Prefix in WH Partition Table	Allows you to choose whether or not to use the prefix partition queries. The prefix is always used with pre-queries.	• (Default) Use prefix in both warehouse partition pre-query and partition query • Use prefix in warehouse partition prequery but not in partition query	(Default) Use prefix in both warehouse partition pre-query and partition query
Distinct/Group by Option (when no aggregation and not table key)	If no aggregation is needed and the attribute defined on the table is not a primary key, tells the SQL Engine whether to use Select Distinct, Group by, or neither.	• Use DISTINCT • No DISTINCT, no GROUP BY • Use GROUP BY	Use DISTINCT
GROUP BY ID Attribute	Determines how to group by a selected ID column when an expression is performed on the ID expression.	• Group by expression • Group by alias • Group by column • Group by position	Group by expression
GROUP BY Non-ID Attribute	Determines how to handle columns for non_ID attributes.	• Use Max • Use Group By	Use Max
Insert Post String	Determines the string that is inserted at the end of insert	User-defined	NULL

Property	Description	Possible Values	Default Value
	and implicit table creation statements.		
Insert Table Option	Determines the string inserted after table name in insert statements; analogous to table option.	User-defined	NULL
Long integer support	Determines whether to map long integers of a certain length as BigInt data types when MicroStrategy creates tables in a database.	• Do not use BigInt • Up to 18 digits • Up to 19 digits	Do not use BigInt
Max Digits in Constant	Sets the maximum number of digits in a constant literal in an insert values statement. (0 = no limit).	User-defined	No limit
Merge Same Metric Expression Option	Determines how to handle metrics that have the same definition.	• Merge same metric expression • Do not merge same metric expression	Merge same metric expression
Select Post String	Defines the custom SQL string to be appended to all `SELECT` statements, for example, FOR FETCH ONLY.	User-defined	NULL
Select Statement Post String	Defines the custom SQL string to be appended to the final `SELECT`.	User-defined	NULL
SQL Hint	This string is placed after the SELECT statement.	User-defined	NULL
SQL Time Format	Determines the format of the time literal accepted in SQL statements.	User-defined	yyyy-mm-dd hh:nn:ss
Timestamp Format	Sets the format of the timestamp literal accepted in the Where clause.	User-defined	yyyy-nn-dd hh:mm:ss
UNION Multiple INSERT	Allows the Analytical Engine to UNION multiple insert statements into the same temporary table.	• Do not use UNION • Use UNION	Do not use UNION
Use Column Type Hint for Parameterized Query	Determines whether the WCHAR data type is used as applicable to return data accurately while using parameterized queries.	• Disabled • Enable ODBC Column Type Binding Hint for “WCHAR” and “CHAR”	Disabled

Attribute Selection and Form Selection Option for Intermediate Passes

Normally, the MicroStrategy SQL Engine selects the minimum number of columns that are needed in each pass. For an intermediate pass, the SQL Engine usually only selects attribute ID forms. The SQL Engine also selects the attributes necessary to make the join, usually key attributes. Then in the final pass, additional attributes or attribute forms that are necessary for report display can be joined.

This algorithm is optimal in most cases, as it minimizes the size of intermediate tables. However, in certain schemas, especially denormalized ones, and schemas that use fact tables as both lookup tables and relationship tables, such an algorithm may cause additional joins in the final pass.

Example

A report template contains the attributes Region and Store, and metrics M1 and M2. M1 uses the fact table FT1, which contains Store_ID, Store_Desc, Region_ID, Region_Desc, and f1. M2 uses the fact table FT2, which contains Store_ID, Store_Desc, Region_ID, Region_Desc, and F2. With the normal SQL Engine algorithm, the intermediate pass that calculates M1 selects Store_ID and F1, the intermediate pass that calculates M2 selects Store_ID and F2. Then the final pass joins these two intermediate tables together. But that is not enough. Since Region is on the template, it should join upward to the region level and find the Region_Desc form. This can be done by joining either FT1 or FT2 in the final pass. So with the original algorithm, either FT1 or FT2 is being accessed twice. If these tables are big, and they usually are, the performance can be very slow. On the other hand, if Store_ID, Store_Desc, Region_ID, and Region_Desc are picked up in the intermediate passes, there is no need to join FT1 or FT2 does not need to be joined in the final pass, thus boosting performance.

For this reason, the following two properties are available in MicroStrategy:

- Attribute Form Selection Option for Intermediate Pass
- Attribute Selection Option for Intermediate Pass

- These properties intend to use bigger (wider) intermediate tables to save additional joins in the final pass and exchange space for time.
- These two properties work independently. One does not influence the other.
- Each property has two values. The default behavior is the original algorithm.
- When the property is enabled:
  - The SQL Engine selects additional attributes or attribute forms in the intermediate pass, when they are directly available.
  - The SQL Engine does not join additional tables to select more attributes or forms. So for intermediate passes, the number of tables to be joined is the same as when the property is disabled.

Attribute Form Selection Option for Intermediate Pass

The Attribute Form Selection Options for Intermediate Pass property determines whether or not the SQL Engine selects the needed attribute forms in the intermediate passes, if available. See the description above for more detailed information.

Levels at which you can set this

Database instance, report, and template

Attribute Selection Option for Intermediate Pass

The Attribute Selection Option for Intermediate Pass property determines whether or not the SQL Engine selects additional attributes (usually parent attributes) needed on the template, other than the needed join ID column in the intermediate passes. See the description above for more detailed information.

Levels at which you can set this

Database instance, report, and template

Bulk Insert String

Bulk Insert String is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Bulk Insert String property appends the string provided in front of the INSERT statement. For Teradata, this property is set to ";" to increase query performance. The string is appended only for the INSERT INTO SELECT statements and not the INSERT INTO VALUES statement that is generated by the Analytical Engine. Since the string is appended for the INSERT INTO SELECT statement, this property takes effect only during explicit, permanent, or temporary table creation.

Levels at which you can set this

Database instance, report, and template

Example

```
Bulk Insert String = ;
```

Constant Column Mode

Constant Column Mode is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Constant Column Mode allows you to choose whether or not to use a GROUP BY and how the GROUP BY should be constructed when working with a column that is a constant. The GROUP BY can be constructed with the column, alias, position numbers, or column expression. Most users do not need to change this setting. It is available to be used with the new Generic DBMS object and if you want to use a different GROUP BY method when working with constant columns.

Levels at which you can set this

Database instance, report, and template

Examples

Pure select, no GROUP BY (default)

```
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, 0 XKYCGT,
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q1_2002 a11
group by a11.QUARTER_ID
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, 1 XKYCGT,
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q2_2002 a11
group by a11.QUARTER_ID
```

Use max, no GROUP BY

```
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, MAX(0) XKYCGT,
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q1_2002 a11
group by a11.QUARTER_ID
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, MAX(1) XKYCGT,
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q2_2002 a11
group by a11.QUARTER_ID
```

GROUP BY column (expression)

```
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, 0 XKYCGT,
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q1_2002 a11
group by a11.QUARTER_ID, 0
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, 1 XKYCGT,
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q2_2002 a11
group by a11.QUARTER_ID, 1
```

GROUP BY alias

```
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, 0 XKYCGT,
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q1_2002 a11
group by a11.QUARTER_ID, XKYCGT
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, 1 XKYCGT,
```

```
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q2_2002 a11
group by a11.QUARTER_ID, XKYCGT
```

GROUP BY position

```
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, 0 XKYCGT,
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q1_2002 a11
group by a11.QUARTER_ID, 2
insert into ZZTP00
select a11.QUARTER_ID QUARTER_ID, 1 XKYCGT,
sum(a11.REG_SLS_DLR) WJXBFS1
from SALES_Q2_2002 a11
group by a11.QUARTER_ID, 2
```

Custom Group Interaction With the Report Filter

The Custom Group Interaction With the Report Filter VLDB property allows you define how a report filter interacts with a custom group.

When a custom group that is created using attribute qualifications is included on a report with a report filter, the report filter is applied to the individual custom group elements. However, if you create a custom group using metric qualifications or banding qualifications, report filters are not applied by default to the custom group elements.

This can cause unexpected results to be returned in some scenarios. For example, a custom group displays revenue totals for customers in columns that represent the range of revenue that the customer is in. A customer that has contributed $7,500 in revenue displays this revenue total in the column for customers that contributed $5,000 to $10,000 in revenue. This custom group is included on a report that has a report filter that restricts results to data for the year 2007 only.

In this scenario, the report filter is evaluated after the custom group. If the same customer that has a total of $7,500 only had $2,500 in 2007, then the report would only display $2,500 for that customer. However, the customer would still be in the $5,000 to $10,000 in revenue range because the custom group did not account for the report filter.

You can define report filter and custom group interaction to avoid this scenario. This VLDB property has the following options:

- **No interaction - static custom group** (default): Report filter qualifications are not applied to custom groups that use metric qualifications or banding qualifications. Filtering is only applied after the custom group has been evaluated.
- **Apply report filter to custom group**: Report filter qualifications are applied to custom groups and are used to determine the values for each custom group element.
- **Apply report filter to custom group, but ignore related elements from the report filter**: Report filter qualifications that do not qualify on attribute elements that are used to define the custom group elements are applied to custom groups. These filter qualifications are used to determine the values for each custom group element. For example, a report filter that qualifies on the Customer attribute is not applied to a custom group that also uses the Customer attribute to define its custom group elements.

For information on custom groups and defining these options for a custom group, see the *Advanced Reporting Guide*.

Levels at which you can set this

Database instance

Data Retrieval Mode

The Data Retrieval Mode VLDB property determines whether data is retrieved using third-party, native APIs. You have the following options:

- **Only ODBC**: Standard methods are used to retrieve data. This option must be used in all cases, except for connections that are expected to make use of the Teradata Parallel Transporter API.
- **Allow Native API**: Third-party native APIs can be used to retrieve data. MicroStrategy supports the use of the Teradata Parallel Transporter API. Enabling Teradata Parallel Transporter can improve performance when retrieving large amounts of data from Teradata, typically 1 Gigabyte and larger, which can occur most commonly in MicroStrategy when publishing Intelligent Cubes.

  Using MicroStrategy Web, you can create a connection to Teradata and import your data. When creating this connection, enabling the Teradata Parallel Transporter options automatically defines this VLDB property as Allow Native API for the connection. For steps to create this type of connection in MicroStrategy Web, see the *Web User Help*.

  You can also select this VLDB property option for the database instance for Teradata connections that are not created through the use of Data Import.

Levels at which you can set this

Database instance and report

Data Retrieval Parameters

The Data Retrieval Mode VLDB property defines the parameters used to retrieve data using third-party, native APIs.

For this VLDB property to take effect, you must defined the Data Retrieval Mode VLDB property (see *Data Retrieval Parameters, page 193*) as Allow Native API. You can then define the required parameters to retrieve data using the third-party, native API. For example, you can enable Teradata Parallel Transporter by defining the following parameters:

- `TD_TDP_ID`: The name or IP address of the machine on which the Teradata data source resides.
- `TD_MAX_SESSIONS`: The maximum number of sessions that can be used to log on to the Teradata database when processing queries in parallel. By default, one session

per Access Process Module (AMP) is used, which is also the maximum number of sessions that can be supported. Type a value to allow fewer sessions than the number of available AMPs.

- `TD_MIN_SESSIONS`: The minimum number of sessions required for the export driver job to complete its processes. The default is one session. This value must be less than or equal to the `TD_MAX_SESSIONS` value.
- `TD_MAX_INSTANCES`: The maximum number of threads that can be used. This option can be defined if the driver has been configured as a master and slave environment that allows for multiple threads. This value must be less than or equal to the `TD_MAX_SESSIONS` value, as a thread can include one or more sessions.
- You can include any additional parameters to apply to the connection. Provide each parameter with the syntax: *`ParameterName=ParameterValue`*.

When providing the parameters and their values, each parameter must be of the form:

`ParameterName=ParameterValue`

Separate each parameter definition with a semicolon (`;`). An example of the full definition of this VLDB property is provided below:

```
TD_TDP_ID=123.45.67.89;TD_MAX_SESSIONS=3;TD_MIN_
SESSIONS=1;TD_MAX_INSTANCES=3
```

Using MicroStrategy Web, you can create a connection to Teradata and import your data. When creating this connection, enabling the Teradata Parallel Transporter options prompts you for this information and automatically updates the VLDB property as required. For steps to create this type of connection in MicroStrategy Web, see the *Web User Help*.

You can also define this VLDB property for the database instance for Teradata connections that are not created through the use of Data Import.

Levels at which you can set this

Database instance and report

Data mart Column Order

This property allows you to determine the order in which data mart columns are created when you configure a data mart from the information in the columns and rows of a report.

You can set this property to either of the following options:

- **Columns created in order based on attribute weight** (default): Data mart columns are created in an order based on their attribute weights. For more information about attribute weights, see *Data mart Column Order, page 194*.
- **Columns created in order in which they appear on the template**: Data mart columns are created in the same order as they appear on the report template.

Levels at which you can set this

Database instance, report, and template

Date Format

The Date Format property specifies the format of the date string literal in the SQL statements when date-related qualifications are present in the report.

Levels at which you can set this

Database instance, report, and template

Example

Default	`yyyy-mm-dd`
Oracle	`dd-mmm-yy`
Teradata	`yyyy/mm/dd`

Date Pattern

Date Pattern is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Date Pattern property is used to add or alter a syntax pattern for handling date columns.

Levels at which you can set this

Database instance, report, and template

Example

Default	No extra syntax pattern for handling dates
Oracle	`To_Date ('#0')`
Tandem	`(d'#0')`

Decimal Separator

The Decimal Separator property specifies whether a "." or "," is used as a decimal separator. This property is used for non-English databases that use commas as the decimal separator.

Levels at which you can set this

Database instance, report, and template

Examples

"." as the decimal separator (default)

```
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
into #ZZTIS00H5K4MQ000
from HARI_COST_STORE_DEP a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.COST_AMT) > 654.357
```

"," as the decimal separator

```
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
into #ZZTIS00H5K5MQ000
from HARI_COST_STORE_DEP a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.COST_AMT) > 654,357
```

Default Attribute Weight

The Default Attribute Weight is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

Use the Default Attribute Weight property to determine how attribute weights should be treated, for those attributes that are not in the attribute weights list.

You can access the attribute weights list from the Project Configuration Editor. In the Project Configuration Editor, expand **Report Definition** and select **SQL generation**. From the Attribute weights section, click **Modify** to open the attribute weights list.

The attribute weights list allows you to change the order of attributes used in the SELECT clause of a query. For example, suppose the Region attribute is placed higher on the attribute weights list than the Customer State attribute. When the SQL for a report containing both attributes is generated, Region is referenced in the SQL before Customer State. However, suppose another attribute, Quarter, also appears on the report template but is not included in the attribute weights list.

In this case, you can select either of the following options within the Default Attribute Weight property to determine whether Quarter is considered highest or lowest on the attribute weights list:

- **Lowest**: When you select this option, those attributes not in the attribute weights list are treated as the lightest weight. Using the example above, with this setting selected, Quarter is considered to have a lighter attribute weight than the other two

attributes. Therefore, it is referenced after Region and Customer State in the SELECT statement.

- **Highest** (default): When you select this option, those attributes not in the attribute weights list are treated as the highest weight. Using the example above, with this setting selected, Quarter is considered to have a higher attribute weight than the other two attributes. Therefore, it is referenced before Region and Customer State in the SELECT statement.

Levels at which you can set this

Database instance only

Disable Prefix in WH Partition Table

The Disable Prefix in WH Partition Table is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This property allows you to provide better support of warehouse partitioning in a distributed database environment.

In a distributed database environment, different tables can have different prefixes. This is also true for partitioning. On one hand, the partition-mapping table (PMT) may have a different prefix from the partition base table (PBT). On the other hand, each PBT may need its own prefix. In MicroStrategy 6.x and earlier, this is achieved by adding one additional column (DDBSOURCE) in the PMT to indicate which table source (prefix) to use. MicroStrategy 7.x and later uses metadata (MD) partitioning and warehouse (WH) partitioning. MD partitioning can handle distributed databases easily, because the metadata contains the PMT as well as the PBT. For WH partitioning, it only has the PMT in the metadata, so it can only set prefixes on the PMT. Currently, this prefix is shared by both the PMT and the PBT. In other words, both the partition prequery (using PMT) and the partition query (using PBT) use the same prefix.

For those projects that need their own prefix in the PBT, the MicroStrategy 6.x approach (using the DDBSOURCE column) no longer works due to architectural changes. The solution is to store the prefix along with the PBT name in the column PBTNAME of the partition mapping table. So instead of storing PBT1, PBT2, and so on, you can put in DB1.PBT1, DB2.PBT2, and so on. This effectively adds a different prefix to different PBTs by treating the entire string as the partition base table name.

The solution above works in most cases but does not work if the PMT needs its own prefix. For example, if the PMT has the prefix “DB0.”, the prequery works fine. However, in the partition query, this prefix is added to what is stored in the PBTNAME column, so it gets DB0.DB1.PBT1, DB0.DB1.PBT2, and so on. This is not what you want to happen. This new VLDB property is used to disable the prefix in the WH partition table. When this property is turned on, the partition query no longer shares the prefix from the PMT. Instead, the PBTNAME column (DB1.PBT1, DB2.PBT2, and so on) is used as the full PBT name.

Even when this property is turned ON, the partition prequery still applies a prefix, if there is one.

Levels at which you can set this

Database instance, report, and template

Distinct/Group by Option (when no aggregation and not table key)

The Distinct/Group by Option property controls the generation of DISTINCT or GROUP BY in the SELECT SQL statement. You can select from the following options:

- **Use DISTINCT** (default)
- **No DISTINCT, no GROUP BY**
- **Use GROUP BY**

If you are using a Vertica database that includes correlated subqueries, to support the use of the Use GROUP By option listed above, you must also define the Sub Query Type VLDB property (see *Optimizing queries, page 135*) to use either of the following options:

  - Use Temporary Table, falling back to EXISTS (SELECT *...) for correlated subquery
  - Use Temporary Table, falling back to IN (SELECT COL) for correlated subquery

Upon selecting an option, a sample SQL statement shows the effect that each option has.

The SQL Engine ignores the option selected for this property in the following situations:

- If there is aggregation, GROUP BY is used without the use of DISTINCT.
- If there is no attribute (only metrics), DISTINCT is not used.
- If there is COUNT (DISTINCT ...) and the database does not support this functionality, a SELECT DISTINCT pass of SQL is used, which is followed by a COUNT(*) pass of SQL.
- If the database does not allow DISTINCT or GROUP BY for certain column data types, DISTINCT and GROUP BY are not used.
- If the select level is the same as the table key level and the table's true key property is selected, DISTINCT is not used.

When none of the above conditions are met, the option selected for this property determines how DISTINCT and GROUP BY are used in the SQL statement.

Levels at which you can set this

Database instance, report, and template

GROUP BY ID Attribute

The GROUP BY ID Attribute is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

This property determines how to group by a selected ID column when an expression is performed on the ID expression. Each of the options is described below.

The code fragment following each description replaces the section named *group by ID* in the following sample SQL statement.

```
select a22.STORE_NBR STORE_NBR,
a22.MARKET_NBR * 10 MARKET_ID,
sum(a21.REG_SLS_DLR) WJXBFS1
from STORE_DIVISION a21
join LOOKUP_STORE a22
on (a21.STORE_NBR = a22.STORE_NBR)
where a22.STORE_NBR = 1
group by a22.STORE_NBR, group by ID
```

The options for this property are:

- **Group by expression** (default): Group by the expression performed in the SELECT statement on the ID column.

  ```
  a22.MARKET_NBR * 10
  ```

- **Group by alias**: Group by the expression alias in the Select statement.

  ```
  MARKET_ID
  ```

- **Group by column**: Group by the column ID, ignoring the expression performed on the ID column.

  ```
  a22.MARKET_NBR
  ```

- **Group by position**: Group by the physical table position of the ID column.

  ```
  2
  ```

Levels at which you can set this

Database instance, report, and template

GROUP BY Non-ID Attribute

The GROUP BY Non-ID Attribute property controls whether or not non-ID attribute forms—like descriptions—are used in the GROUP BY. If you do not want non-ID columns in the GROUP BY, you can choose to use a MAX when the column is selected so that it is not used in the GROUP BY.

Levels at which you can set this

Database instance, report, and template

Examples

Use Max (default)

```
select a11.MARKET_NBR MARKET_NBR,
                max(a14.MARKET_DESC) MARKET_DESC,
                a11.CLASS_NBR CLASS_NBR,
                max(a13.CLASS_DESC) CLASS_DESC,
                a12.YEAR_ID YEAR_ID,
                max(a15.YEAR_DESC) YEAR_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from MARKET_CLASS a11
                join LOOKUP_DAY a12
                  on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
                join LOOKUP_CLASS a13
                  on (a11.CLASS_NBR = a13.CLASS_NBR)
                join LOOKUP_MARKET a14
                  on (a11.MARKET_NBR = a14.MARKET_NBR)
                join LOOKUP_YEAR a15
                  on (a12.YEAR_ID = a15.YEAR_ID)
group by a11.MARKET_NBR, a11.CLASS_NBR,
a12.YEAR_ID
```

Use Group by

```
select a11.MARKET_NBR MARKET_NBR,
                a14.MARKET_DESC   MARKET_DESC,
                a11.CLASS_NBR CLASS_NBR,
                a13.CLASS_DESC CLASS_DESC,
                a12.YEAR_ID YEAR_ID,
                a15.YEAR_DESC YEAR_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from MARKET_CLASS a11
                join LOOKUP_DAY a12
                  on (a11.CUR_TRN_DT = a12.CUR_TRN_DT)
                join LOOKUP_CLASS a13
                  on (a11.CLASS_NBR = a13.CLASS_NBR)
                join LOOKUP_MARKET a14
                  on (a11.MARKET_NBR = a14.MARKET_NBR)
                join LOOKUP_YEAR a15
                  on (a12.YEAR_ID = a15.YEAR_ID)
group by a11.MARKET_NBR,
                a14.MARKET_DESC,
                a11.CLASS_NBR,
                a13.CLASS_DESC,
                a12.YEAR_ID,
                a15.YEAR_DESC
```

Insert Post String

The Insert Post String property allows you to define a custom string to be inserted at the end of the INSERT statements.

The # character is a special token that is used in various patterns and is treated differently than other characters. One single # is absorbed and two # are reduced to a single #. For example to show three # characters in a statement, enter six # characters in the code. You can get any desired string with the right number of # characters. Using the # character is the same as using the ; character.

Levels at which you can set this

Database instance, report, and template

Example

```
Insert into TABLENAME
select A1.COL1, A2.COL2, A3.COL3
from TABLE1 A1, TABLE2 A2, TABLE3 A3
where A1.COL1=A2.COL1 and A2.COL4=A3.COL5 */Insert
Post String/*
```

Insert Table Option

The Insert Table Option property allows you to define a custom string to be inserted after the table name in the insert statements. This is analogous to table option.

The # character is a special token that is used in various patterns and is treated differently than other characters. One single # is absorbed and two # are reduced to a single #. For example to show three # characters in a statement, enter six # characters in the code. You can get any desired string with the right number of # characters. Using the # character is the same as using the ; character.

Levels at which you can set this

Database instance, report, and template

Example

```
Insert into TABLENAME */Insert Table Option/*
select A1.COL1, A2.COL2, A3.COL3
from TABLE1 A1, TABLE2 A2, TABLE3 A3
where A1.COL1 = A2.COL1 and A2.COL4=A3.COL5
```

Long integer support

Long integer support is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

With this VLDB property you can determine whether long integers are mapped to a BigInt data type when MicroStrategy creates tables in the database. A data mart is an

example of a MicroStrategy feature that requires MicroStrategy to create tables in a database.

When long integers from databases are integrated into MicroStrategy, the Big Decimal data type is used to define the data in MicroStrategy. Long integers can be of various database data types such as Number, Decimal, and BigInt.

In the case of BigInt, when data that uses the BigInt data type is integrated into MicroStrategy as a Big Decimal, this can cause a data type mismatch when MicroStrategy creates a table in the database. MicroStrategy does not use the BigInt data type by default when creating tables. This can cause a data type mismatch between the originating database table that contained the BigInt and the database table created by MicroStrategy.

You can use the following VLDB settings to support BigInt data types:

- **Do not use BigInt** (default): Long integers are not mapped as BigInt data types when MicroStrategy creates tables in the database. This is the default behavior.

  If you use BigInt data types, this can cause a data type mismatch between the originating database table that contained the BigInt and the database table created by MicroStrategy.

- **Up to 18 digits**: Long integers that have up to 18 digits are converted into BigInt data types.

  This setting is a good option if you can ensure that your BigInt data uses no more than 18 digits. The maximum number of digits that a BigInt can use is 19. With this option, if your database contains BigInt data that uses all 19 digits, it is not mapped as a BigInt data type when MicroStrategy creates a table in the database.

  However, using this setting requires you to manually modify the column data type mapped to your BigInt data. You can achieve this by creating a column alias for the column of data in the Attribute Editor or Fact Editor in MicroStrategy. The column alias must have a data type of Big Decimal, a precision of 18, and a scale of zero. For steps to create a column alias to modify a column data type, see the Project Design Guide.

- **Up to 19 digits**: Long integers that have up to 19 digits are converted into BigInt data types.

  Using this option enables BigInt data that uses up to 19 digits to be correctly mapped as a BigInt data types when MicroStrategy creates tables in the database. This option does not require you to create a column alias.

  However, this option can cause an overflow error if you have long integers that use exactly 19 digits, and its value is greater than the maximum allowed for a BigInt (9,223,372,036,854,775,807).

Levels at which you can set this

Database instance, report, and template

Max Digits in Constant

Max Digits in Constant is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Max Digits in Constant property controls the number of significant digits that get inserted into columns during Analytical Engine inserts. This is only applicable to real numbers and not to integers.

Levels at which you can set this

Database instance only

Examples

Database-specific setting

SQL Server	28
Teradata	18

Max Const Digits = 0

```
Insert into #ZZTIS00H6WQMD001 values (4, 339515.0792)
```

Max Const Digits = 2

```
Insert into #ZZTIS00H6WTMD001 values (4, 33)
```

Max Const Digits = 7

```
Insert into #ZZTIS00H6WVMD001 values (4, 339515.0)
```

Merge Same Metric Expression Option

The Merge Same Metric Expression Option VLDB property allows you to determine whether the SQL Engine should merge metrics that have the same definition, or whether it should process the metrics separately. If you do not want metrics with identical definitions to be merged, select **Do not merge same metric expression**.

Levels at which you can set this

Database instance, report, and template

Select Post String

The Select Post String property allows you to define a custom string to be inserted at the end of all `SELECT` statements generated by the Analytical Engine.

To include a post string only on the final SELECT statement you should use the Select Statement Post String VLDB property, which is described in *Select Post String, page 203*.

Levels at which you can set this

Database instance, report, and template

Example

The SQL statement shown below displays an example of where the Select Post String and Select Statement Post String VLDB properties would include their SQL statements.

```
with gopa1 as
(select a12.REGION_ID REGION_ID
        from CITY_CTR_SLS a11
                join LU_CALL_CTR a12
                on (a11.CALL_CTR_ID = a12.CALL_CTR_ID)
        group by a12.REGION_ID
        having sum(a11.TOT_UNIT_SALES) = 7.0
/* select post string */)select
        a11.REGION_ID REGION_ID,
        a14.REGION_NAME REGION_NAME0,
        sum(a11.TOT_DOLLAR_SALES) Revenue
from STATE_SUBCATEG_REGION_SLS a11
        join gopa1 pa12
        on (a11.REGION_ID = pa12.REGION_ID)
        join LU_SUBCATEG a13
        on (a11.SUBCAT_ID = a13.SUBCAT_ID)
        join LU_REGION a14
        on (a11.REGION_ID = a14.REGION_ID)
where a13.CATEGORY_ID in (2)
group by a11.REGION_ID,
        a14.REGION_NAME/* select post string */
           /* select statement post string */
```

Select Statement Post String

The Select Statement Post String VLDB property allows you to define a custom SQL string to be inserted at the end of the final SELECT statement.

This can be helpful if you use common table expressions with an IBM DB2 database. These common table expressions do not support certain custom SQL strings. This VLDB property allows you to apply the custom SQL string to only the final SELECT statement which does not use a common table expression.

Levels at which you can set this

Database instance, report, and template

Example

The SQL statement shown below displays an example of where the Select Post String and Select Statement Post String VLDB properties include their SQL statements.

```
with gopa1 as
(select a12.REGION_ID REGION_ID
        from CITY_CTR_SLS a11
                join LU_CALL_CTR a12
                on (a11.CALL_CTR_ID = a12.CALL_CTR_ID)
        group by a12.REGION_ID
        having sum(a11.TOT_UNIT_SALES) = 7.0
/* select post string */)select
        a11.REGION_ID REGION_ID,
        a14.REGION_NAME REGION_NAME0,
        sum(a11.TOT_DOLLAR_SALES) Revenue
from STATE_SUBCATEG_REGION_SLS a11
        join gopa1 pa12
        on (a11.REGION_ID = pa12.REGION_ID)
        join LU_SUBCATEG a13
        on (a11.SUBCAT_ID = a13.SUBCAT_ID)
        join LU_REGION a14
        on (a11.REGION_ID = a14.REGION_ID)
where a13.CATEGORY_ID in (2)
group by a11.REGION_ID,
        a14.REGION_NAME/* select post string */
           /* select statement post string */
```

SQL Hint

The SQL Hint property is used for the Oracle SQL Hint pattern. This string is placed after the SELECT word in the Select statement. This property can be used to insert any SQL string that makes sense after the SELECT in a Select statement, but it is provided specifically for Oracle SQL Hints.

Levels at which you can set this

Database instance, report, and template

Example

```
SQL Hint = /* FULL */
Select /* + FULL */ A1.STORE_NBR,
max(A1.STORE_DESC)
From LOOKUP_STORE A1
Where A1.STORE_NBR = 1
Group by A1.STORE_NBR
```

SQL Time Format

The SQL Time Format property allows you to determine the format of the time literal accepted in SQL statements. This is a database-specific property; some examples are shown in the table below.

Example

Database Type	Time Format
Default	`yyyy-mm-dd hh:nn:ss`
Microsoft SQL Server	`mm/dd/yyyy hh:nn:ss`
Oracle	`mm/dd/yyyy hh:nn:ss`
Sybase IQ	`hh:nn:ss:lll`

Levels at which you can set this

Database instance, template, and report

Timestamp Format

The Timestamp Format property allows you to determine the format of the timestamp literal accepted in the WHERE clause. This is a database-specific property; some examples are shown in the table below.

Levels at which you can set this

Database instance, report, and template

Example

Default	`yyyy-mm-dd hh:nn:ss`
DB2	`yyyy-mm-dd-hh.nn.ss`
RedBrick	`mm/dd/yyyy hh:nn:ss`

UNION Multiple INSERT

The Union Multiple Insert property allows the Analytical Engine to UNION multiple INSERT statements into the same temporary table. This is a database-specific property. Some databases do not support the use of Unions.

Levels at which you can set this

Database instance, report, and template

Databases automatically set to Use Union

- DB2 UDB

- SQL Server
- Teradata

Use Column Type Hint for Parameterized Query

Use Column Type Hint for Parameterized Query is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Use Column Type Hint for Parameterized Query VLDB property determines whether the WCHAR data type is used when applicable to return data accurately while using parameterized queries. This VLDB property has the following options:

- **Disabled** (default): This option is recommended unless you are encountering the data inconsistencies described below.
- **Enable ODBC Column Type Binding Hint for "WCHAR" and "CHAR"**: This option should be used only if you have enabled parameterized queries in MicroStrategy for your database and data is not being correctly displayed on reports. This can include viewing question marks in place of other valid characters. This can occur for Netezza databases.

  By selecting this option, the WCHAR data type is used when applicable so that the data is returned correctly while using parameterized queries.

Levels at which you can set this

Database instance only

Creating and supporting tables with SQL: Tables

The table below summarizes the Tables VLDB properties that are available. Additional details about each property, including examples where necessary, are provided in the sections following the table.

Property	Description	Possible Values	Default Value
Alias Pattern	Used to alter the pattern for aliasing column names. Automatically set for Microsoft Access users.	User-defined	AS
Attribute ID Constraint	Defines the column constraints (for example, NULL or NOT NULL) put on the ID form of attributes.	User-defined	NULL
Character Column Option and National Character Column Option	Defines how to support multiple character sets used in Teradata.	User-defined	NULL

Property	Description	Possible Values	Default Value
Column Pattern	Used to alter the pattern for column names.	User-defined	#0.[#1]
Commit After Final Drop	Determines whether to issue a COMMIT statement after the final DROP statement	• No Commit after the final Drop statement • Commit after the final Drop statement	No Commit after the final Drop statement
Commit Level	Sets when to issue a COMMIT statement after creating an intermediate table.	• No Commit • Post DDL • Post DML • Post DDL and DML	No Commit
CREATE and INSERT support	Defines whether MicroStrategy can perform `CREATE` and `INSERT` statements against the database for a database instance.	• CREATE and INSERT statements are supported • CREATE and INSERT statements are not supported	CREATE and INSERT statements are supported
Create Post String (see *Table Prefix, Table Qualifier, Table Option, Table Descriptor, Table Space, & Create Post String*)	Defines the string appended after the CREATE TABLE statement.	User-defined	NULL
Drop Temp Table Method	Determines when to drop an intermediate object.	• Drop after final pass • Do nothing • Truncate table then drop after final pass	Drop after final pass
Fallback Table Type	Determines the type of table that is generated if the Analytical Engine cannot generate a derived table or common table.	• Permanent table • True temporary table • Fail report	Permanent table
Hexadecimal Character Transformation	Allows string characters to be converted into specific character encoding required for some Unicode implementations.	• Do not apply hexadecimal character transformation to quoted strings • Apply hexadecimal	Do not apply hexadecimal character transformation to quoted strings

Property	Description	Possible Values	Default Value
		character transformation to quoted strings of all character types • Apply hexadecimal character transformation to quoted strings of type NChar and NVarChar	
Intermediate Table Type	Determines the type of intermediate (temp) table to create.	• Permanent table • Derived table • Common table expression • True temporary table • Temporary view	Permanent table
Maximum SQL Passes Before FallBack	Determines how many passes are allowed for a report that uses intermediate tables. If a report exceeds this limit, the table type defined by the Fallback Table Type VLDB property is used for the report.	User-defined	No limit
Maximum Tables in FROM Clause Before FallBack	Determines how many tables in a single FROM clause are allowed for a report that uses intermediate tables. If a report exceeds this limit, the table type defined by the Fallback Table Type VLDB property is used for the report.	User-defined	No limit
National Character Column Option	Defines how to support multiple character sets used in Teradata.	User-defined	NULL
Parallel SQL Execution Intermediate Table Type	Determines the type of intermediate table created when parallel query execution is used.	• Permanent Table • Derived Table with Fallback Table Type as Permanent Table	Permanent Table
Table Creation Type	Determines the method to	• Explicit table	Explicit table

Property	Description	Possible Values	Default Value
	create an intermediate table.	• Implicit table	
Table Descriptor (see *Table Prefix, Table Qualifier, Table Option, Table Descriptor, Table Space, & Create Post String*)	Defines the string to be placed after the word TABLE in the CREATE TABLE statement.	User-defined	NULL
Table Option (see *Table Prefix, Table Qualifier, Table Option, Table Descriptor, Table Space, & Create Post String*)	Defines the string to be placed after the table name in the CREATE TABLE statement.	User-defined	NULL
Table Prefix (see *Table Prefix, Table Qualifier, Table Option, Table Descriptor, Table Space, & Create Post String*)	Defines the string to be added to a table name, for example, CREATE TABLE prefix.Tablename. (See Note below.)	User-defined	NULL
Table Qualifier (see *Table Prefix, Table Qualifier, Table Option, Table Descriptor, Table Space, & Create Post String*)	Defines the key words placed immediately before "table." For example, CREATE volatile Table.	User-defined	NULL
Table Space (see *Table Prefix, Table Qualifier, Table Option, Table Descriptor, Table Space, & Create Post String*)	String appended after the CREATE TABLE Statement but before any Primary Index/Partition key definitions. (See Note below.)	User-defined	NULL

To populate dynamic information by the Analytical Engine, insert the following syntax into Table Prefix and Table Space strings:

!d inserts the date.

!o inserts the report name.

!u inserts the user name.

Alias Pattern

Alias Pattern is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Alias Pattern property allows you to alter the pattern for aliasing column names. Most databases do not need this pattern, because their column aliases simply follow the column name with only a space between them. However, Microsoft Access needs an AS between the column name and the given column alias. This pattern is automatically set for Microsoft Access users. This property is provided for customers using the Generic DBMS object because some databases may need the AS or another pattern for column aliasing.

Levels at which you can set this

Database instance only

Attribute ID Constraint

This property is available at the attribute level. You can access this property by opening the Attribute Editor, selecting the **Tools** menu, then choosing **VLDB Properties.**

When creating intermediate tables in the explicit mode, you can specify the NOT NULL/NULL constraint during the table creation phase. This takes effect only when permanent or temporary tables are created in the explicit table creation mode. Furthermore, it applies only to the attribute columns in the intermediate tables.

Levels at which you can set this

Database instance and attribute

Example

NOT NULL

```
create table ZZTIS003HHUMQ000 (
        DEPARTMENT_NBR NUMBER(10, 0) NOT NULL,
        STORE_NBR NUMBER(10, 0) NOT NULL)
```

Character Column Option and National Character Column Option

The Character Column Option and National Character Column Option VLDB properties allow you to support the character sets used in Teradata. Teradata allows character sets to be defined on a column-by-column basis. For example, one column in Teradata may use a Unicode character set, while another column uses a Latin character set.

MicroStrategy uses two sets of data types to support multiple character sets. The Char and VarChar data types are used to support a character set. The NChar and NVarChar data types are used to support a different character set than the one supported by Char

and VarChar. The NChar and NVarChar data types are commonly used to support the Unicode character set while Char and VarChar data types are used to support another character set.

You can support the character sets in your Teradata database using these VLDB properties:

- The Character Column Option VLDB property defines the character set used for columns that use the MicroStrategy Char or VarChar data types. If left empty, these data types use the default character set for the Teradata database user.

  You can define a specific data type by typing `CHARACTER SET CHARACTER_SET_NAME`, where *`CHARACTER_SET_NAME`* is the name of the character set. For example, `CHARACTER SET LATIN` defines MicroStrategy's Char and VarChar data types to support the Latin character set.

  This character set definition is included in SQL statements as shown in the example below:

  ```
  CREATE TABLE text_fields (Text_Field1 VARCHAR(10)
  CHARACTER SET LATIN,Text_Field2 VARCHAR(10) CHARACTER
  SET LATIN,)
  ```

- The National Character Column Option VLDB property defines the character set used for columns that use the MicroStrategy NChar or NVarChar data types. If left empty, these data types use the default character set for the Teradata database user.

  You can define a specific data type by typing `CHARACTER SET CHARACTER_SET_NAME`, where *`CHARACTER_SET_NAME`* is the name of the character set. For example, `CHARACTER SET UNICODE` defines MicroStrategy's NChar and NVarChar data types to support the Unicode character set.

  If you use the Unicode character set and it is not the default character set for the Teradata database user, you should define NChar and NVarChar data types to use the Unicode character set.

  This character set definition is included in SQL statements as shown in the example below:

  ```
  CREATE TABLE text_fields (Text_Field1 VARCHAR(10)
  CHARACTER SET UNICODE,Text_Field2 VARCHAR(10) CHARACTER
  SET UNICODE,)
  ```

For example, your Teradata database uses the Latin and Unicode character sets, and the default character set for your Teradata database is Latin. In this scenario you should leave Character Column Option empty so that it uses the default of Latin. You should also define National Character Column as `CHARACTER SET UNICODE` so that NChar and NVarChar data types support the Unicode data for your Teradata database.

To extend this example, assume that your Teradata database uses the Latin and Unicode character sets, but the default character set for your Teradata database is Unicode. In this scenario you should leave National Character Column Option empty so that it uses the default of Unicode. You should also define Character Column as `CHARACTER SET LATIN` so that Char and VarChar data types support the Latin data for your Teradata database.

The Character Column Option and National Character Column Option VLDB properties can also support the scenario where two character sets are used, and Unicode is not one of these character sets. For this scenario, you can use these two VLDB properties to define which MicroStrategy data types support the character sets of your Teradata database.

Levels at which you can set this

Database instance only

Column Pattern

Column Pattern is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Column Pattern property allows you to alter the pattern for column names. Most databases do not need this pattern altered. However, if you are using a case-sensitive database and need to add double quotes around the column name, this property allows you to do that.

Levels at which you can set this

Database instance only

Example

The standard column pattern is `#0.#1`. If double quotes are needed, the pattern changes to:

```
"#0.#1"
```

Commit After Final Drop

The Commit After Final Drop property determines whether or not to issue a COMMIT statement after the final DROP statement.

Levels at which you can set this

Database instance and report

Commit Level

The Commit Level property is used to issue COMMIT statements after the Data Definition Language (DDL) and Data Manipulation Language (DML) statements. When this property is used in conjunction with the INSERT MID Statement, INSERT PRE

Statement, or TABLE POST Statement VLDB properties, the COMMIT is issued before any of the custom SQL passes specified in the statements are executed. The only DDL statement issued after the COMMIT is issued is the explicit CREATE TABLE statement. Commit is issued after DROP TABLE statements even though it is a DDL statement.

The only DML statement issued after the COMMIT is issued is the INSERT INTO TABLE statement. If the property is set to Post DML, the COMMIT is not issued after an individual INSERT INTO VALUES statement; instead, it is issued after all the INSERT INTO VALUES statements are executed.

The Post DDL COMMIT only shows up if the Intermediate Table Type VLDB property is set to Permanent tables or Temporary tables and the Table Creation Type VLDB property is set to Explicit mode.

The Post DML COMMIT only shows up if the Intermediate Table Type VLDB property is set to Permanent tables, Temporary tables, or Views.

Not all database platforms support COMMIT statements and some need special statements to be executed first, so this property must be used in projects whose warehouse tables are in databases that support it.

Levels at which you can set this

Database instance, report, and template

Examples

Table Creation Type is set to Explicit

No Commit (default)

```
create table ZZTIS00H8L8MQ000 (
                DEPARTMENT_NBR          NUMBER(10, 0),
                STORE_NBR NUMBER(10, 0)) tablespace users
insert into ZZTIS00H8L8MQ000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
from HARI_STORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_DEPARTMENT a11,
                ZZTIS00H8L8MQ000 pa1,
                HARI_LOOKUP_DEPARTMENT a12,
                HARI_LOOKUP_STORE a13
where a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
                a11.STORE_NBR = pa1.STORE_NBR and
                a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR and
                a11.STORE_NBR = a13.STORE_NBR
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Post DDL Commit

```
create table ZZTIS00H8LHMQ000 (
                DEPARTMENT_NBR NUMBER(10, 0),
                STORE_NBR NUMBER(10, 0)) tablespace users
commit
insert into ZZTIS00H8LHMQ000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
from HARI_STORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_DEPARTMENT a11,
                ZZTIS00H8LHMQ000 pa1,
                HARI_LOOKUP_DEPARTMENT a12,
                HARI_LOOKUP_STORE a13
where a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
                a11.STORE_NBR = pa1.STORE_NBR and
                a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR and
                a11.STORE_NBR = a13.STORE_NBR
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Post DDL & Post DML Commit

```
create table ZZTIS00H8LZMQ000 (
                DEPARTMENT_NBR NUMBER(10, 0),
                STORE_NBR NUMBER(10, 0)) tablespace users
commit
insert into ZZTIS00H8LZMQ000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
from HARI_STORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
commit
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_DEPARTMENT a11,
                ZZTIS00H8LZMQ000 pa1,
                HARI_LOOKUP_DEPARTMENT a12,
                HARI_LOOKUP_STORE a13
where a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
                a11.STORE_NBR = pa1.STORE_NBR and
                a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR and
                a11.STORE_NBR = a13.STORE_NBR
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Table Creation Type is set to Implicit

No Commit (default)

```
create table ZZTIS00H8LCMQ000 tablespace users as
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
```

```
                a11.STORE_NBR STORE_NBR
from HARI_STORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_DEPARTMENT a11,
                ZZTIS00H8LCMQ000 pa1,
                HARI_LOOKUP_DEPARTMENT a12,
                HARI_LOOKUP_STORE a13
where a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
                a11.STORE_NBR = pa1.STORE_NBR and
                a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR and
                a11.STORE_NBR = a13.STORE_NBR
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Post DDL Commit

```
create table ZZTIS00H8LLMQ000 tablespace users as
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
from HARI_STORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_DEPARTMENT a11,
                ZZTIS00H8LLMQ000 pa1,
                HARI_LOOKUP_DEPARTMENT a12,
                HARI_LOOKUP_STORE a13
where a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
                a11.STORE_NBR = pa1.STORE_NBR and
                a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR and
                a11.STORE_NBR = a13.STORE_NBR
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Post DML Commit

```
create table ZZTIS00H8LTMQ000 tablespace users as
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
from HARI_STORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
commit
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_DEPARTMENT a11,
                ZZTIS00H8LTMQ000 pa1,
                HARI_LOOKUP_DEPARTMENT a12,
                HARI_LOOKUP_STORE a13
where a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
```

```
                a11.STORE_NBR = pa1.STORE_NBR and
                a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR and
                a11.STORE_NBR = a13.STORE_NBR
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Post DDL & Post DML Commit

```
create table ZZTIS00H8M3MQ000 tablespace users as
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
from HARI_STORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
commit
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HARI_STORE_DEPARTMENT a11,
                ZZTIS00H8M3MQ000 pa1,
                HARI_LOOKUP_DEPARTMENT a12,
                HARI_LOOKUP_STORE a13
where a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
                a11.STORE_NBR = pa1.STORE_NBR and
                a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR and
                a11.STORE_NBR = a13.STORE_NBR
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

CREATE and INSERT support

The CREATE and INSERT support VLDB property defines whether MicroStrategy can perform `CREATE` and `INSERT` statements against the database for a database instance. This VLDB property has the following options:

- **CREATE and INSERT statements are supported** (default): Allows MicroStrategy to perform `CREATE` and `INSERT` statements against the database for a database instance. These statements are required for various MicroStrategy features. This setting is required for the primary database instance and for databases that are required to support data mart reports. For information on primary database instances, see the *Installation and Configuration Guide*.

  This setting is recommended for databases that are used to support fully functioning MicroStrategy projects.

- **CREATE and INSERT statements are not supported**: MicroStrategy is prohibited from performing `CREATE` and `INSERT` statements against the database for a database instance. This option can be used if the database that you connect to is meant to only act as a repository of information that cannot be modified from within MicroStrategy.

  This option can also be used along with the MultiSource Option feature, which allows you to access multiple databases in one MicroStrategy project. You can define your secondary database instances to disallow `CREATE` and `INSERT` statements so

that all information is only inserted into the primary database instance. For information on the MultiSource Option feature, see the *Project Design Guide*.

You can also use this option to avoid the creation of temporary tables on databases for various performance or security purposes.

This option does not control the SQL that can be created and executed against a database using Freeform SQL and Query Builder reports.

Levels at which you can set this

Database instance only

Drop Temp Table Method

The Drop Temp Table Method property specifies whether the intermediate tables, permanent tables, temporary tables, and views are to be dropped at the end of report execution. Dropping the tables can lock catalog tables and affect performance, so dropping the tables manually in a batch process when the database is less active can result in a performance gain. The trade-off is space on the database server. If tables are not dropped, the tables remain on the database server using space until the database administrator drops them.

This VLDB property also allows you to truncate intermediate tables, permanent tables, temporary tables, and views prior to dropping them.

Levels at which you can set this

Database instance, report, and template

Fallback Table Type

All reports can be resolved using permanent or temporary intermediate tables. Generating derived tables, common table expressions, and views as a means of resolving reports is also provided. Derived tables, common table expressions, and views cannot cover all the scenarios. For example, they cannot be used when the report contains Analytical Engine SQL, partitioning, and certain cases of outer joins. In such a scenario, the MicroStrategy SQL Engine needs a fallback mechanism provided by the Fallback Table Type property. If the Intermediate Table Type VLDB property (described below) is set to Derived Table or Common Table Expression or Views, and the SQL Engine concludes that the report cannot be resolved using that setting, it reads the Fallback Table Type VLDB property and resolves the report by generating **Permanent tables** or **Temporary tables** according to the option that you set.

However, there may be scenarios where you do not want to create permanent tables or temporary tables to support these types of reports. To prevent the creation of permanent or temporary tables, you can set the Fallback Table Type VLDB property to **Fail report**. This causes reports that rely on the Fallback Table Type to fail, so it should only be used when it is necessary to prevent the creation of permanent or temporary tables.

Levels at which you can set this

Database instance, report, and template

Hexadecimal Character Transformation

The Hexadecimal Character Transformation property is only relevant when you are using a Unicode Teradata database for the data warehouse. Most databases do not need this property, because the ODBC driver handles the conversion automatically. In some Unicode databases, to process SQL containing character strings inside quotations, those characters must be converted to hexadecimal representation. Turning this property on means characters within quoted strings are converted into hexadecimal using UTF-8 encoding.

Levels at which you can set this

Database instance only

Examples

Do not apply hexadecimal character transformation to quoted strings (default)

```
insert into mytable values ('A')
```

Apply hexadecimal character transformation to quoted strings

```
insert into mytable values ('4100'XCV)
```

Where 4100 is the hexadecimal representation of the character “A” using UTF-8 Unicode encoding.

Intermediate Table Type

The Intermediate Table Type property specifies what kinds of intermediate tables are used to execute the report. All reports can be executed using permanent and temporary tables. There are certain scenarios involving partitioning, outer joins, and analytical functions that the report cannot execute using derived tables, common table expressions, or views. If this is the case, the Fallback Table Type VLDB property (described above) is used to execute the report. The temporary table syntax is specific to each platform.

This property can have a major impact on the performance of the report. Permanent tables are usually less optimal. Derived tables, common table expressions, and true temporary tables usually perform well, but they do not work in all cases and for all databases. The default setting is permanent tables, because it works for all databases in all situations. However, based on your database type, this setting is automatically changed to what is generally the most optimal option for that platform, although other options could prove to be more optimal on a report-by-report basis. You can access the VLDB Properties Editor for the database instance for your database (see *Opening the VLDB*

Properties Editor, page 19), and then select the **Use default inherited value** check box to determine the default option for your database.

To help support the use of common table expressions and derived tables, you can also use the Maximum SQL Passes Before FallBack and Maximum Tables in FROM Clause Before FallBack VLDB properties. These properties (described in *Maximum SQL Passes Before FallBack, page 222* and *Maximum Tables in FROM Clause Before FallBack, page 223*) allow you to define when a report is too complex to use common table expressions and derived table expressions and instead use a fallback table type.

In cases where queries are performed in parallel (through the use of *Optimizing queries, page 135*) the intermediate table type is determined by the VLDB property *Parallel SQL Execution Intermediate Table Type, page 224*.

Levels at which you can set this

Database instance, report, and template

Examples

The following is an output from a DB2 UDB 7.x project.

Permanent Table (default)

```
create table ZZIS03CT00 (
                DEPARTMENT_NBR DECIMAL(10, 0),
                STORE_NBR DECIMAL(10, 0))
insert into ZZIS03CT00
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
        a11.STORE_NBR STORE_NBR
from HSTORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
        a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
        max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
        a11.STORE_NBR STORE_NBR,
        max(a13.STORE_DESC) STORE_DESC,
        sum(a11.TOT_SLS_DLR) TOTALSALES
from HSTORE_DEPARTMENT a11
        join ZZIS03CT00 pa1
           on (a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
        a11.STORE_NBR = pa1.STORE_NBR)
        join HLOOKUP_DEPARTMENT a12
           on (a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR)
        join HLOOKUP_STORE a13
           on (a11.STORE_NBR = a13.STORE_NBR)
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Derived Table

```
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
        max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
        a11.STORE_NBR STORE_NBR,
        max(a13.STORE_DESC) STORE_DESC,
        sum(a11.TOT_SLS_DLR) TOTALSALES
from HSTORE_DEPARTMENT a11
        join (select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
```

```
                a11.STORE_NBR STORE_NBR
        from HSTORE_DEPARTMENT a11
        group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
        having sum(a11.TOT_SLS_DLR) > 100000
        ) pa1
           on (a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
        a11.STORE_NBR = pa1.STORE_NBR)
        join HLOOKUP_DEPARTMENT a12
           on (a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR)
        join HLOOKUP_STORE a13
           on (a11.STORE_NBR = a13.STORE_NBR)
group by a11.DEPARTMENT_NBR,
        a11.STORE_NBR
```

Common Table Expression

```
with pa1 as
 (select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                        a11.STORE_NBR STORE_NBR
                from HSTORE_DEPARTMENT a11
                group by a11.DEPARTMENT_NBR,
                        a11.STORE_NBR
                having sum(a11.TOT_SLS_DLR) > 100000
                )
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HSTORE_DEPARTMENT a11
        join pa1
           on (a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
        a11.STORE_NBR = pa1.STORE_NBR)
        join HLOOKUP_DEPARTMENT a12
           on (a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR)
        join HLOOKUP_STORE a13
           on (a11.STORE_NBR = a13.STORE_NBR)
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Temporary Table

```
declare global temporary table session.ZZIS03CU00(
                DEPARTMENT_NBR DECIMAL(10, 0),
                STORE_NBR DDECIMAL(10, 0))
                        on commit preserve rows not logged
insert into session.ZZIS03CU00
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
from HSTORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HSTORE_DEPARTMENT a11
        join session.ZZIS03CU00 pa1
           on (a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
               a11.STORE_NBR = pa1.STORE_NBR)
        join HLOOKUP_DEPARTMENT a12
           on (a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR)
        join HLOOKUP_STORE a13
```

```
           on (a11.STORE_NBR = a13.STORE_NBR)
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
```

Views

```
create view ZZIS03CV00 (DEPARTMENT_NBR, STORE_NBR) as
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                a11.STORE_NBR STORE_NBR
from HSTORE_DEPARTMENT a11
group by a11.DEPARTMENT_NBR,
                a11.STORE_NBR
having sum(a11.TOT_SLS_DLR) > 100000
select a11.DEPARTMENT_NBR DEPARTMENT_NBR,
                max(a12.DEPARTMENT_DESC) DEPARTMENT_DESC,
                a11.STORE_NBR STORE_NBR,
                max(a13.STORE_DESC) STORE_DESC,
                sum(a11.TOT_SLS_DLR) TOTALSALES
from HSTORE_DEPARTMENT a11
        join ZZIS03CV00 pa1
           on (a11.DEPARTMENT_NBR = pa1.DEPARTMENT_NBR and
               a11.STORE_NBR = pa1.STORE_NBR)
        join HLOOKUP_DEPARTMENT a12
           on (a11.DEPARTMENT_NBR = a12.DEPARTMENT_NBR)
        join HLOOKUP_STORE a13
           on (a11.STORE_NBR = a13.STORE_NBR)
group by a11.DEPARTMENT_NBR, a11.STORE_NBR
```

Maximum SQL Passes Before FallBack

The Maximum SQL Passes Before FallBack VLDB property allows you to define reports to use common table expressions or derived tables while also using temporary or permanent tables for complex reports.

Using common table expressions or derived tables can often provide good performance for reports. However, some production environments have shown better performance when using temporary tables for reports that require multi-pass SQL.

To support the use of the best table type for each type of report, you can use the Maximum SQL Passes Before FallBack VLDB property to define how many passes are allowed for a report that uses intermediate tables. If a report uses more passes than are defined in this VLDB property, the table type defined in the Fallback Table Type VLDB property (see *Fallback Table Type, page 218*) is used rather than the table type defined in the Intermediate Table Type VLDB property (see *Intermediate Table Type, page 219*).

For example, you define the Intermediate Table Type VLDB property to use derived tables for the entire database instance. This default is then used for all reports within that database instance. You also define the Fallback Table Type VLDB property to use temporary tables as the fallback table type. For your production environment, you define the Maximum SQL Passes Before FallBack VLDB property to use the fallback table type for all reports that use more than five passes.

A report is executed. The report requires six passes of SQL to return the required report results. Usually this type of report would use derived tables, as defined by the Intermediate Table Type VLDB property. However, since it uses more passes than the limit defined in the Maximum SQL Passes Before FallBack VLDB property, it must use the fallback table type. Since the Fallback Table Type VLDB property is defined as

temporary tables, the report uses temporary tables to perform the multi-pass SQL and return the report results.

Levels at which you can set this

Database instance, report, and template

Maximum Tables in FROM Clause Before FallBack

The Maximum Tables in FROM Clause Before FallBack VLDB property allows you to define more reports to use common table expressions or derived tables while also using temporary or permanent tables for complex reports.

Using common table expressions or derived tables can often provide good performance for reports. However, some production environments have shown better performance when using temporary tables for reports that require joining a large amount of database tables.

To support the use of the best table type for each type of report, you can use the Maximum Tables in FROM Clause Before FallBack VLDB property (see *Fallback Table Type, page 218*) to define how many tables are allowed in a `From` clause for a report that uses intermediate tables. If a report uses more tables in a `From` clause than are defined in this VLDB property, the table type defined in the Fallback Table Type VLDB property is used rather than the table type defined in the Intermediate Table Type VLDB property (see *Intermediate Table Type, page 219*).

For example, you define the Intermediate Table Type VLDB property to use derived tables for the entire database instance. This default is then used for all reports within that database instance. You also define the Fallback Table Type VLDB property to use temporary tables as the fallback table type. For your production environment, you define the Maximum Tables in FROM Clause Before FallBack VLDB property to use the fallback table type for all reports that use more than seven tables in a `From` clause.

A report is executed. The report requires a SQL statement that includes nine tables in the `From` clause. Usually this type of report would use derived tables, as defined by the Intermediate Table Type VLDB property. However, since it uses more tables in the `From` clause than the limit defined in the Maximum Tables in FROM Clause Before FallBack VLDB property, it must use the fallback table type. Since the Fallback Table Type VLDB property is defined as temporary tables, the report uses temporary tables to perform the SQL statement and return the report results.

Levels at which you can set this

Database instance, report, and template

National Character Column Option

For a description of this VLDB property, see *Character Column Option and National Character Column Option, page 211*.

Levels at which you can set this

Database instance only

Parallel SQL Execution Intermediate Table Type

Parallel SQL Execution Intermediate Table Type is an advanced property that is hidden by default. For information on how to display this property, see *Viewing and changing advanced VLDB properties, page 22*.

The Parallel SQL Execution Intermediate Table Type property determines the type of intermediate table that is used when Parallel Query Execution (see *Optimizing queries, page 135*) is employed for reports and Intelligent Cubes. If Parallel Query Execution is not enabled, or the queries cannot be processed in parallel, the intermediate table type is determined by the VLDB property *Intermediate Table Type, page 219*.

This VLDB property has the following options:

- **Permanent Table**: When the queries for a report or Intelligent Cube are performed in parallel, any intermediate tables are created as permanent tables. This provides broad support as all databases can support permanent tables.
- **Derived Table with Fallback Table Type as Permanent Table**: When the queries for a report or Intelligent Cube are performed in parallel, any intermediate tables are created as derived tables. This can improve performance for databases that support derived tables. However, not all databases support derived tables. Refer to your third-party database vendor documentation to determine if your database supports derived tables.

  If you select this option and derived tables cannot be created for your database, permanent tables are created instead.

Levels at which you can set this

Database instance, report, and template

Table Creation Type

The Table Creation Type property tells the SQL Engine whether to create table implicitly or explicitly. Some databases do not support implicit creation, so this is a database-specific setting.

Levels at which you can set this

Database instance, report, and template

Examples

Explicit table (default)

```
create table TEMP1 (
STORE_NBR INTEGER,
TOT_SLS DOUBLE,
PROMO_SLS DOUBLE)
insert into TEMP1
select a21.STORE_NBR STORE_NBR,
(sum(a21.REG_SLS_DLR) + sum(a21.PML_SLS_DLR)) TOT_SLS,
sum(a21.PML_SLS_DLR) PROMO_SLS
from STORE_DIVISION a21
where a21.STORE_NBR = 1
group by a21.STORE_NBR
```

Implicit table

```
create table TEMP1 as
select a21.STORE_NBR STORE_NBR,
(sum(a21.REG_SLS_DLR) + sum(a21.PML_SLS_DLR)) TOT_SLS,
sum(a21.PML_SLS_DLR) PROMO_SLS
from STORE_DIVISION a21
where a21.STORE_NBR = 1
group by a21.STORE_NBR
```

Table Prefix, Table Qualifier, Table Option, Table Descriptor, Table Space, & Create Post String

These properties can be used to customize the CREATE TABLE SQL syntax for any platform. All of these properties are reflected in the SQL statement only if the Intermediate Table Type VLDB property is set to Permanent Table. Customizing a CREATE TABLE statement is only possible for a permanent table. For all other valid Intermediate Table Type VLDB settings, the SQL does not reflect the values set for these properties. The location of each property in the CREATE TABLE statement is given below.

```
create /* Table Qualifier */ table /*Table
Descriptor*//* Table Prefix */ZZTIS003RB6MD000 /*Table
Option*/ (
        STORE_NBR NUMBER,
        CLEARANCESAL DOUBLE)
/* Table Space */
/* Create PostString */
```

For platforms like Teradata and DB2 UDB 6.x and 7.x versions, the Primary Index or the Partition Key SQL syntax is placed between the Table Space and Create Post String VLDB property.

Levels at which you can set this

Database instance, report, and template

Default VLDB settings for specific data sources

MicroStrategy certifies and supports connection and integration with many third-party databases, MDX cube sources, and other data sources.

These include databases, data sources, and MDX cube sources from third-party vendors such as IBM DB2, Oracle, Informix, SAP, Sybase, Microsoft, Netezza, Teradata, and so on. For certification information on these data sources, refer to the *MicroStrategy Readme*.

Certain VLDB properties use different default settings depending on which data source you are using. This allows MicroStrategy to both properly support and take advantage of certain characteristics of each third-party data source.

You can determine the default options for each VLDB property for a database by performing the steps below. This provides an accurate list of default VLDB properties for your third-party data source for the version of MicroStrategy that you are using.

Prerequisites

- You have a user account with administrative privileges.
- Ensure that you have fully upgraded your MicroStrategy environment and the available database types, as described in *Upgrading the VLDB options for a particular database type, page 25*.

To create a list of default VLDB settings for a data source

1. In Developer, log in to a project source using an account with administrative privileges.
2. From the **Folder List**, expand **Administration**, then **Configuration Managers**, and select **Database Instances**.
3. From the **File** menu, point to **New**, and select **Database Instance**. The Database Instances Editor opens.
4. In the **Database instance name** field, type a descriptive name for the database instance.
5. From the **Database connection type** drop-down list, select the appropriate option for the data source to list default VLDB settings for. For example, you can select Oracle 11g to determine the default VLDB settings for an Oracle 11g database.

   To return a list of default VLDB properties for a data source, only an appropriate database connection type needs to be defined for the database instance; a connection to a data source does not need to be made. After you create the list of default VLDB settings for the data source, you can delete the database instance or modify it to connect to your data source.
6. Click **OK** to exit the Database Instances Editor and save the database instance.
7. Right-click the new database instance that you created and select **VLDB Properties**. The VLDB Properties Editor opens.
8. From the **Tools** menu, ensure that **Show Advanced Settings** is selected.
9. From the **Tools** menu, select **Create VLDB Settings Report**. The VLDB Settings Report dialog box opens.

A VLDB settings report can be created to display current VLDB settings for database instances, attributes, metrics, and other objects in your project. For information on creating a VLDB settings report for other purposes, see *Creating a VLDB settings report, page 20*.

10. Select the **Show descriptions of setting values** check box. This displays the descriptive information of each default VLDB property setting in the VLDB settings report.
11. The VLDB settings report now displays all the default settings for the data source. You can copy the content in the report using the **Ctrl+C** keys on your keyboard, then paste the information into a text editor or word processing program (such as Microsoft Word) using the **Ctrl+V** keys.
12. Once you are finished reviewing and copying the VLDB settings report, click the close button to close the VLDB Settings Report dialog box.
13. From the **File** menu, select **Close** to close the VLDB Properties Editor.
14. You can then either delete the database instance that you created earlier, or modify it to connect to your data source.

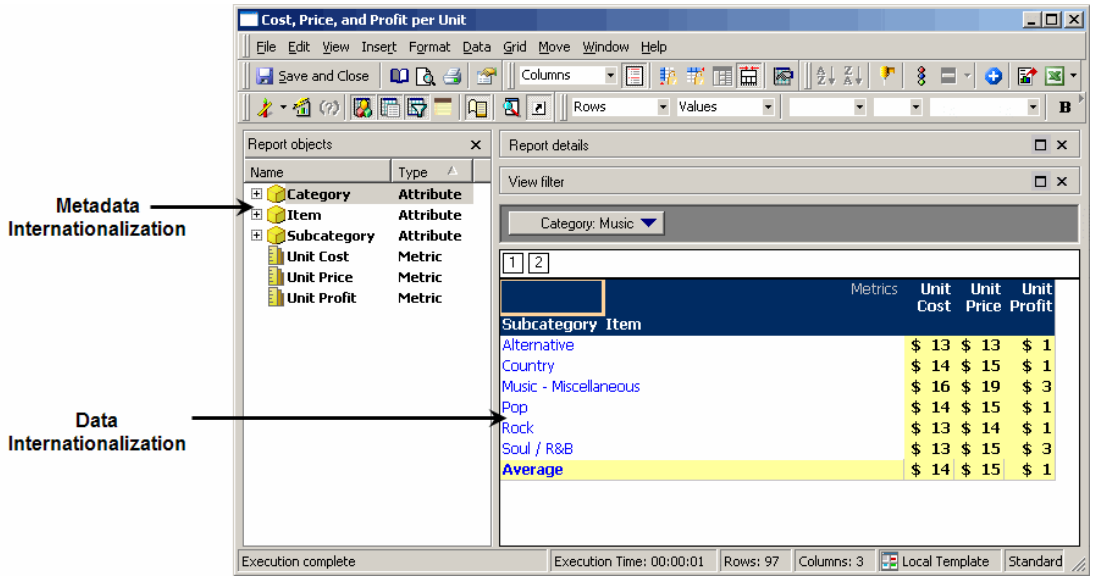
2

CREATING A MULTILINGUAL ENVIRONMENT: INTERNATIONALIZATION

This chapter shows you how to use MicroStrategy to internationalize a project in your MicroStrategy environment, to make it available to a multilingual audience. This includes internationalizing data in your data warehouse and metadata objects in the MicroStrategy metadata repository. This chapter also shows you how to display a translated MicroStrategy interface.

Translating your data and metadata allows your users to view their reports in a variety of languages. It also allows report designers and others to display report and document editors and other objects editors in various languages. And because all translation information can be stored in the same project, project maintenance is easier and more efficient for administrators.

The image below shows which parts of a report are translated using data internationalization and which parts of a report are translated using metadata internationalization:



This chapter assumes you have an understanding of standard MicroStrategy metadata objects, as well as how your organization stores translated data in your data warehouse system.

This chapter includes the following information:

- *About internationalization, page 229* provides an introduction to internationalization in MicroStrategy, with examples; it also provides information on how caching works in an internationalized environment.
- *Best practices for implementing internationalization, page 231*
- *Preparing a project to support internationalization, page 232* provides steps to take during installation or upgrade to prepare your projects for internationalization.
- *Providing metadata internationalization, page 235* explains how the metadata can be internationalized.
- *Providing data internationalization, page 244* provides steps to connect to, set up, and store translated data within your data warehouse so that it can be retrieved and displayed in MicroStrategy reports.
- *Making translated data available to users, page 251* describes the hierarchy of preferences that a user can have set, and how that hierarchy works.
- *Achieving the correct language display, page 265* provides a table of the functionality that MicroStrategy users can access to take advantage of internationalization.
- *Maintaining your internationalized environment, page 269* provides information on using scripts with Command Manager to automate your internationalized environment; moving translated objects between projects; adding languages to be supported by a project; adding a custom language; and applying security to your internationalized environment, including creating specialized translator user roles.

About internationalization

For a fully internationalized environment, both metadata internationalization and data internationalization are required. However, you can internationalize only your metadata, or only your data, based on your needs. Both are described below.

This section also describes translating the user interface and how internationalization affects report/document caching.

About metadata internationalization

Metadata internationalization displays translated object strings based on a user's locale and other language preferences in MicroStrategy, for objects that are stored in the MicroStrategy metadata, such as metric names and report names. For example, you have two metrics stored in your metadata repository, named Cost and Profit. These metadata objects will appear on reports accessed by both English and Italian users. You can use metadata internationalization to configure MicroStrategy to automatically display Cost and Profit to the English users and Metrica costo and Metric proffito to the Italian users.

Metadata internationalization (or MDI) involves exporting object strings to a location where they can be translated, performing the linguistic translation, and importing the newly translated object strings back into the metadata repository. You can also translate individual objects one at a time, using the Object Translation Editor.

For steps to perform these procedures, see *Providing metadata internationalization, page 235*.

About data internationalization

Data internationalization allows a single report definition to contain different attribute forms for different languages available to users, based on a user's locale and other language preferences in MicroStrategy. For example, you want to display a product name stored in your data warehouse to two different users, one who reads English and one who reads French. Both users execute and view the same product report. You can use data internationalization to configure MicroStrategy to automatically display A Tale of Two Cities to the English user and Un Conte de Deux Villes to the French user.

Data internationalization (or DI) involves configuring your data warehouse so that tables and other structures allow MicroStrategy to access data in the appropriate language for the user requesting the report. If you use multiple warehouses to store translated data, DI involves connecting MicroStrategy to the appropriate warehouses.

Depending on the data internationalization model you choose, which is based on the structure of your translation storage environment (as described above), you may only be able to translate the DESC (description) form.

For steps to perform these procedures, see *Providing data internationalization, page 244*.

About internationalizing the general user interface

The MicroStrategy general user interface (such as the File menu, Edit menu, and so on) can also be displayed in various languages. This translation process is not part of metadata or data internationalization, but steps to select a preferred interface language are part of this chapter. MicroStrategy provides translated strings for the general user interface in several languages. You can display the MicroStrategy general user interface in a selected language using the **MicroStrategy Developer Preferences** options in Developer and the **Preferences** link in MicroStrategy Web:

- For steps to select the interface language in Developer, see *Selecting the Interface language preference, page 253*.
- For steps to select the interface language in Web, click **Help** in MicroStrategy Web.

Caching and internationalization

For details about caching, see the *Caching* chapter in the *MicroStrategy System Administration Guide*.

Object caching is not affected by internationalization.

Normal report and document caching behavior is not affected, regardless of the types of internationalization that you implement. Specifically, data internationalization methods

(SQL-based and connection-based, both described below) do not affect standard report and document caching behavior.

Different caches are created for different DI languages, but not for different MDI languages. When a user whose MDI language and DI language are French runs a report, a cache is created containing French data and using the report's French name. When a second user whose MDI language and DI language are German runs the same report, a new cache is created with German data and using the report's German name. If a third user whose MDI language is French and DI language is German runs the same report, the second user's cache is hit. Two users with the same DI language preference use the same cache, regardless of MDI preferences.

A report's data internationalization language is displayed in a Data Language column in the Cache Monitor. This helps the administrator identify the difference between cached reports, when it is important to be able to identify these differences.

Best practices for implementing internationalization

- Make sure your database supports the character set(s) that are required by the various languages you intend to support in your MicroStrategy project. MicroStrategy recommends using a Unicode database to ensure all your languages are supported. For details, see *Adding or removing a language in the system, page 270*.
- If you will be supporting double-byte languages (such as Japanese or Korean), make sure that appropriate fonts are available for graph labels, text fields in documents, and so on. Appropriate fonts to support double-byte languages are generally Unicode fonts. An example of an effective Unicode font for double-byte languages is Arial Unicode MS. Most Unicode fonts ensure that all characters can be displayed correctly when a report or document is displayed in a double-byte language.

  
  Not all Unicode fonts can display double-byte languages, for example, Lucida Sans Unicode does not display double-byte languages.

- All SQL-based qualifications contained in a given report should be in a single language. SQL-based qualifications include such things as report filters, metrics, and prompts.
- If you have old projects with metadata objects that have been previously translated, it is recommended that you merge your translated strings from your old metadata into the newly upgraded metadata using MicroStrategy Project Merge. For steps, see *Translating already translated pre-9.x projects, page 243*.
- It is recommended for Developer internationalization that you use a unified locale. For example, if French is the language selected for the interface, the metadata objects language preference and report data language preference, as well as number and date preferences, should also be in French.
- If you are using or plan to use MicroStrategy Intelligent Cubes, and you plan to implement data internationalization, it is recommended that you use a SQL-based DI model. The SQL-based DI model is described in *Providing data internationalization,*

page 244. Because a single Intelligent Cube cannot connect to more than one data warehouse, using a connection-based DI model requires a separate Intelligent Cube to be created for each language, which can be resource-intensive. Details on this cost-benefit analysis as well as background information on Intelligent Cubes are in the *MicroStrategy In-memory Analytics Guide*.

Preparing a project to support internationalization

The procedures in this section will help you modify existing MicroStrategy projects to support both metadata and data internationalization. These procedures perform several important modifications to your metadata, including making it Unicode-compliant, providing some new translations for system objects, and other project-level preparations.

These procedures must be performed whether you plan to support only metadata internationalization, only data internationalization, or both.

Required procedures in this section include:

- *Adding internationalization tables to the metadata repository, page 232*
- *Updating your project's metadata definitions, page 233*

Prerequisites

- This chapter includes steps to be taken when installing or upgrading to the latest version of Developer. You should be prepared to use the steps below during the installation or upgrade process. For steps to install, see the *Installation and Configuration Guide*. For steps to upgrade, see the *Upgrade Guide*.

Adding internationalization tables to the metadata repository

The first step to internationalizing your data and metadata is to add the internationalization tables to your MicroStrategy metadata repository.

This step must be performed before you update your project's metadata definitions.

This step must be completed during your installation or upgrade to the latest version of Developer. For steps to install, see the *Installation and Configuration Guide*. For steps to perform a general MicroStrategy upgrade, see the *Upgrade Guide*.

To add internationalization tables to the metadata repository

1. During the upgrade or installation process, select **Upgrade existing environment to MicroStrategy Analytics Enterprise** in the Configuration Wizard, and click **Next**.

2 Continue working through the steps in the *Installation and Configuration Guide* or the *Upgrade Guide* to complete the process.

Updating your project's metadata definitions

After you add internationalization tables to your metadata repository as described above, you must update your project's metadata with the latest definitions.

This procedure may have been completed during your installation or upgrade to the latest version of Developer. If it was not part of the install or upgrade, it must be performed to support metadata and data internationalization. For steps to install, see the *Installation and Configuration Guide*. For steps to upgrade, see the *Upgrade Guide*.

To update metadata definitions

1 In Developer, double-click the name of the project that you want to internationalize.

2 Log into the project. You are prompted to update your project. Click **Yes**.

The metadata is updated to the latest version of MicroStrategy.

Updating system object translations

This optional procedure lets you "automatically translate" system objects such as folder names, security roles, and user groups, by accessing translations that come with MicroStrategy for those objects.

If you prefer to provide your own translations (for example if you will be customizing folder names), you do not need to perform this procedure.

This procedure may have been completed during your installation or upgrade to the latest version of Developer. For steps to install, see the *Installation and Configuration Guide*. For steps to upgrade, see the *Upgrade Guide*.

For projects created before MicroStrategy version 8.x, due to changes in folder structure it is possible that system objects cannot be updated if they have been renamed.

To update system object translations

1 Reload the project before updating system object translations. To do this, in the Folder List on the left, within the appropriate project source, expand **Administration**, expand **System Administration**, and select **Projects**. Right-click the project, point to **Administer project**, and click **Unload**. After the project unloads, click **Load**.

2 Right-click the project you have upgraded, and select **Project Configuration**. The Project Configuration Editor opens.

3 Expand **Project Definition**, expand **Update**, then select **Translations**.

4 Click **Update**. System object translations are updated and displayed when users apply a language preference.

Allowing access to languages and language objects

Internationalization for languages and language objects is controlled primarily through access control lists (ACLs). You can allow permissions to specific users for each object that needs to be translated, or for each language object (an object that represents a language in your system).

Access to add or modify a translation

You can create a specialized user account for a translator that restricts their access in MicroStrategy to only translating objects into a specific language. For steps, see *Creating translator roles, page 274*.

By default, administrators and object owners can translate an object or modify an existing translation. Use ACLs to provide other users Write access to an object, if other users need to translate that object. To change ACL permissions, right-click the object and select **Properties**, then select **Security** on the left. For details on each ACL and what access it allows, click **Help**.

You can also provide a user with the Use Repository Translation Wizard privilege. This allows a user to perform the necessary steps to translate or modify translations of strings in all languages, without giving the user the ability to modify an object in any other way. To change a privilege, open the user in the User Editor and select **Project Access** on the left, then expand the Object Manager set of privileges on the right and select the **Use Repository Translation Wizard** check box.

Access to select or enable displayed languages: Language objects

By default, MicroStrategy users are provided with appropriate privileges to Browse and Use language objects, such that analysts can select a language as their display preference if that language has been enabled for a project. Project administrators can enable any languages available in the system.

You can modify these default privileges for a specific user role or a specific language object.

To modify access to a language object

1 In the Folder List on the left, within the appropriate project source, expand **Administration**.

2 Expand **Configuration Managers**, then select **Languages**.

3 All language objects are listed on the right. To change ACL permissions for a language object, right-click the object and select **Properties**.

4 Select **Security** on the left. For details on each ACL and what access it allows, click **Help**.

Providing metadata internationalization

Metadata internationalization (MDI) displays translated object strings based on a user's locale and other language preferences in the software, for objects that are stored in the MicroStrategy metadata, such as metric names, report names, the Public Objects system folder, security role names, user group names, and so on. Metadata translation also includes embedded text strings (embedded in an object's definition), such as prompt instructions, aliased names (which can be used in attributes, metrics, and custom groups), consolidation element names, custom group element names, graph titles, and threshold text.

Metadata object translation does not include configuration objects (such as the user object), function names, data mart table names, and so on.

Begin metadata translation by enabling languages for your project's metadata objects; see *Enabling and disabling metadata languages, page 235*. Then use the appropriate set of procedures below, depending on whether translations already exist for your project or you will be translating your project for the first time:

- *Translating your project for the first time, page 238*
- *Translating already translated pre-9.x projects, page 243*

Enabling and disabling metadata languages

To support the display of translations for metadata object names and descriptions, you must enable languages for your project. The languages you enable are those languages you want to support for that project.

You can also disable languages for a project.

Enabling metadata languages while creating a new project

If you plan to provide an internationalized project, you can enable internationalization when creating a new project. For information on the structure of your data warehouse to support internationalization for a new project, and steps to enable internationalization while creating a new project, see the *Project Design Guide*.

Enabling metadata languages for an existing project

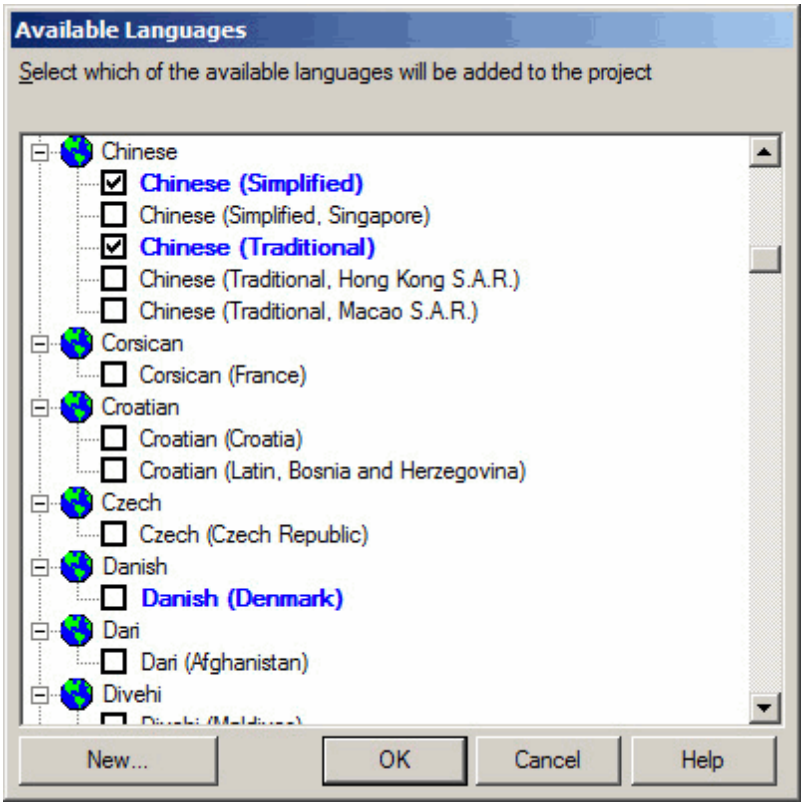
After the metadata has been updated and your project has been prepared for internationalization (usually performed during the MicroStrategy installation or upgrade), you enable languages so they will be supported by the project for metadata internationalization.

Prerequisites

- Gather a list of languages used by filters and prompts in the project. These languages should be enabled for the project, otherwise a report containing a filter or prompt in a language not enabled for the project will not be able to execute successfully.

To enable metadata languages for a project

1. Log into the project as a user with Administrative privileges.
2. Right-click the project and select **Project Configuration**. The Project Configuration Editor opens.
3. On the left side of the Project Configuration Editor, expand **Language** and select **Metadata**.
4. Click **Add** to see a list of available languages. The Available Languages dialog box opens, as shown below:



The languages displayed in bold blue are those languages that the metadata objects have been enabled to support. This list is displayed as a starting point for the set of languages you can choose to enable for supporting data internationalization.

To add a new language, click **New**. The Languages Editor opens. For steps to create a custom language, see ***Adding or removing a language in the system, page 270***.

5 Select the check boxes for the languages that you want to enable for this project.

- Enabled languages will appear in the Repository Translation Wizard for string translation, as well as in Developer's My Preferences and Web's Preferences, for users to select their own preferred language for the project.
- Reports that contain filters or prompts in a translated language will execute successfully if the project has that language enabled.

6 Click **OK.** The Available Languages Dialog Box closes.

7 Select one of the languages on the right side to be the default language for this project. The default language is used by the system to maintain object name uniqueness.

This may have been set when the project was first created. If so, it will not be available to be selected here.

Once the project default language is set, it cannot be changed unless you duplicate the project and change the default language of the duplicated project. Individual objects within a project can have their default language changed.

If you are enabling a language for a project that has been upgraded from 8.x or earlier, the default metadata language must be the language in which the project was originally created (the 8.x Developer language at the time of project creation). Be sure to select the default language that matches the language selected when the project was originally created. You can then add other languages to support the project. To change a project's default language, you must duplicate the project and change the default language in the duplicated project.

8 Click **OK**.

9 Disconnect and reconnect to the project source.

10 Update the out-of-the-box MicroStrategy metadata objects. To do this, in Developer, right-click the project and select **Project Configuration**. Expand **Project definition**, expand **Update**, select **Translations**, and click **Update**.

Disabling metadata languages for a project

You can use the steps below to disable a language for a project. When a language has been disabled from a project, that language is no longer available for users to select as a language preference, and the language cannot be seen in any translation-related interfaces, such as an object's Translation dialog box.

If a user's preferred language is disabled, the next lower priority language preference will take effect. To see the language preference priority hierarchy, see *Configuring metadata object and report data language preferences, page 254*.

Any translations for the disabled language are not removed from the metadata with these steps. Retaining the translations in the metadata allows you to enable the language again later, and the translations will still exist. To remove translations in the disabled language

from the metadata, objects that contain these terms must be modified individually and saved.

To disable metadata languages in a project

1. Log in to a project as a user with administrative privileges.
2. Right-click the project and select **Project Configuration**. The Project Configuration Editor opens.
3. On the left side of the Project Configuration Editor, expand **Language**, then select **Metadata**.
4. On the right side, under Selected Languages, clear the check box for the language that you want to disable for the project, and click **OK.** The Project Configuration Editor closes.

Translating your project for the first time

Translating a project involves providing translated strings for metadata object names and descriptions.

If you use translator roles, be sure to assign the appropriate permissions and privileges in MicroStrategy to your translators before beginning the translation steps. See *Creating translator roles, page 274* for details.

There are two methods to translate metadata objects, depending on whether you want to translate a large number of objects or just one or two objects:

- **Translate a large number of objects**: Extract strings in bulk to a translation database, translate them, and import them back into MicroStrategy. The MicroStrategy Repository Translation Wizard is the recommended method to internationalize your metadata objects. Steps to access this tool are below.
- **Translate one or more objects in a folder**: Right-click the object and select **Translate**. Type the translated word(s) for each language this object supports, and click **OK**. To translate several objects, select them all while holding Shift or Ctrl, then right-click and select **Translate**. For details to use the Object Translation dialog box, click **Help**.

The rest of this section describes the method to translate bulk object strings, using a separate translation database, with the Repository Translation Wizard.

The Repository Translation Wizard does not support translation of configuration objects (such as the user object). It does support object descriptors, including embedded text. These are detailed in the introduction to *Providing metadata internationalization, page 235*.

If your project has not yet been translated, metadata internationalization involves the following high-level steps:

All of the procedures in this section assume that your projects have been prepared for internationalization. Preparation steps are in *Preparing a project to support internationalization, page 232*.

1. Enable languages for the metadata repository (see *Enabling and disabling metadata languages, page 235*)
2. Export object strings to a location where they can be translated (see *Extracting metadata object strings for translation, page 239*)
3. Perform the linguistic translation (see *Translating metadata object strings in the translation database, page 239*)
4. Import the newly translated object strings back into the metadata repository (see *Importing translated strings from the translation database to the metadata, page 242*)

Extracting metadata object strings for translation

The MicroStrategy Repository Translation Wizard supports Microsoft Access and Microsoft SQL Server databases as translation repositories. The translation repository is where strings are extracted to and where the actual translation process is performed.

You cannot extract strings from the project's default metadata language.

It is recommended that objects are not modified between the extraction process and the import process. This is especially important for objects with location-specific strings: attribute aliases, metric aliases, custom group elements, and document text boxes.

To extract a large number of object strings for translation

1. Open the Repository Translation Wizard. To do this, from the **Start** menu, point to **All Programs**, then **MicroStrategy Tools**, then select **Repository Translation Wizard**. The Repository Translation Wizard opens.
2. Click **Next** to begin.
3. Click **Help** for details on each option in each page of the wizard.

   To extract strings from the metadata, select the Export Translations option from the Metadata Repository page in the wizard.

Translating metadata object strings in the translation database

The extraction process performed by the Repository Translation Wizard creates a table in the translation database, with the following columns:

- PROJECTID: This is the ID of the project from which the string is extracted.
- OBJECTID: This is the ID of the object from which the string is extracted.

- OBJECTTYPE: Each object is associated with a numeric code. For example, documents are represented by OBJECTTYPE code 55.
- EMBEDDEDID: An embedded object is an object contained inside another object, for example, a metric object that is part of a report object. If the string is extracted from an embedded object, the ID of this embedded object is stored in this column. The value 0 indicates that the string is not extracted from an embedded object.
- EMBEDDEDTYPE: This is a numeric representation of the type of the embedded object. The value 0 indicates that the string is not extracted from an embedded object.
- UNIQUEKEY: This is a key assigned to the extracted string to identify the string within the object.
- READABLEKEY: This is a description of the extracted string within the object, for example, Prompt Title, Prompt Description, Object Name, Template Subtotal Name, and so on. The READABLEKEY is a readable form of the UNIQUEKEY.
- LOCALEID: This indicates the language of the extracted string in the TRANSLATION column.

  MicroStrategy uses locale IDs to uniquely identify languages. For consistency, MicroStrategy uses the same locale IDs as Microsoft. The following table lists the language codes for the languages that MicroStrategy supports out-of-the-box.

Language	Language Code
Chinese (Simplified)	2052
Chinese (Traditional)	1028
English (US)	1033
French (France)	1036
German (Germany)	1031
Italian (Italy)	1040
Japanese	1041
Korean	1042
Portuguese (Brazil)	1046
Spanish (Spain)	3082
Swedish	1053

For custom languages, MicroStrategy assigns a unique language ID based on the base language that it is derived from.

- TRANSLATION: This is the column where the extracted string is stored.
- TRANSVERSIONID: This is the version ID of the object at the time of export.

- REFTRANSLATION: This column contains the extracted string in the translation reference language, which is selected by the user from the Repository Translation Wizard during export.

  This string is used only as a reference during the translation process. For example, if the translator is comfortable with the German language, you can set German as the translation reference language. The REFTRANSLATION column will then contain all the extracted strings in the German language, for the translator to use as a reference when she is translating extracted strings.

  If no reference language string is available, the string from the object's primary language is exported so that this column is not empty for any string.

- STATUS: You can use this column to enter flags in the table to control which strings are imported back into the metadata. A flag is a character you type, for example, a letter, a number, or a special character (as long as it is allowed by your database). When you use the wizard to import the strings back into the metadata, you can identify this character for the system to use during the import process, to determine which strings to import.

  For example, if a translator has finished only some translations, you may want to import only the completed ones. Or if a reviewer has completed the language review for only some of the translations, you may wish to import only those strings that were reviewed. You can flag the strings that were completed and are ready to be imported.

- OBJVERSIONID: This is the version ID of objects at the time of import.

- SYNCHFLAG: This is a system flag and is automatically generated during import. The following values are used:

  - 0: This means that the object has not been modified between extraction and import.
  - 1: This means that the object has been modified between extraction and import.
  - 2: This means that the object that is being imported is no longer present in the metadata.

  System flags are automatically applied to strings during the import process, so that you can view any string-specific information in the log file.

- LASTMODIFIED: This is the date and time when the strings were extracted.

Once the extraction process is complete, the strings in the translation database need to be translated in the extraction table described above. This is generally performed by a dedicated translation team or a 3rd party translation vendor.

- If an object has an empty translation in a user's chosen project language preference, the system defaults to displaying the object's default language, so it is not necessary to add translations for objects that are not intended to be translated. For details on language preferences, see *Selecting preferred languages for interfaces, reports, and objects, page 252*.
- If you performed a Search for Objects in the Repository Translation Tool, you may notice that the number of rows in the extraction table might not match the number of rows returned in the search results. This is because a search returns all objects that meet the search requirements; the search does not filter for only those items that can be translated. Thus, for example, the search may return a row for the lookup table LU_YEAR, but the extraction process does not extract the LU_YEAR string because there is no reason to translate a lookup table's name. To determine whether an object's name can be translated, right-click the object, select **Properties**, and look for the **International** option on the left. If this option is missing, the object is not supported for translation.

To confirm that your translations have successfully been imported back into the metadata, navigate to one of the translated objects in Developer, right-click, and select Properties. On the left, select International, then click Translate. The table shows all translations currently in the metadata for this object.

Importing translated strings from the translation database to the metadata

After strings have been translated by a language expert in the translation database, they must be re-imported into the MicroStrategy metadata.

To import translated strings

1. Open the Repository Translation Wizard. To do this, from the **Start** menu, point to **All Programs**, then **MicroStrategy Tools**, then select **Repository Translation Wizard**. The Repository Translation Wizard opens.
2. Click **Next** to begin.
3. Click **Help** for details on each option in each page of the wizard.

   To import strings from the translation database back into the metadata, select the **Import Translations** option from the Metadata Repository page in the wizard.

After the strings are imported back into the project, any objects that were modified while the translation process was being performed, are automatically marked with a **1**. These translations should be checked for correctness, since the modification may have included changing the object's name or description.

When you are finished with the string translation process, you can proceed with data internationalization if you plan to provide translated report data to your users. For

background information and steps, see *Providing data internationalization, page 244*. You can also set user language preferences for translated metadata objects and data in *Enabling or disabling languages in the project to support DI, page 248*.

Translating already translated pre-9.x projects

You may have your translated information spread out among several individual, monolingual projects and you want to add multilingual support and combine them into a single, all-inclusive multilingual project called a master project.

Even if you maintain separate production projects in separate languages, the ideal scenario is to create a single development project where translations are maintained for all languages that are required by any regional production projects.

If you use translator roles, be sure to assign the appropriate permissions and privileges in MicroStrategy to your translators before beginning the translation steps. See *Creating translator roles, page 274* for details.

When translated projects already exist, metadata internationalization involves the following high-level steps:

All of the procedures in this section assume that you have completed any final import of translations to your pre-9.x project using the old Repository Translation Tool, and that your projects have been prepared for internationalization. Preparation steps are in *Preparing a project to support internationalization, page 232*.

1. Enable languages for the metadata repository (see *Enabling and disabling metadata languages, page 235*). For the master project, be sure to enable all languages that you will be supporting.
2. Back up your existing translated strings by extracting all objects from the old translated projects using the MicroStrategy Repository Translation Wizard (see *Extracting metadata object strings for translation, page 239*).
3. Merge the translated projects into the master project using the Project Merge Wizard. Do not merge any translations.

   You now have a single master project that contains all objects that were present in both the original master project and in the translated project.
4. Extract all objects from the master project using the MicroStrategy Repository Translation Wizard (see *Extracting metadata object strings for translation, page 239*).
5. Provide translations for all objects in the translated language (see *Translating metadata object strings in the translation database, page 239*).
6. Import all translations back into the master project (see *Importing translated strings from the translation database to the metadata, page 242*).
7. After translation verification, duplicate the master project so that you have a development project, a testing project, and at least one production project.

Providing data internationalization

Data internationalization (or DI) allows you to display translated report and document results to users from your data warehouse. Data internationalization allows a single report definition to contain different attribute elements for each language available to users, with the appropriate element displayed based on the user's locale and other language preferences in the software.

Data internationalization involves the following high-level steps:

> All of the procedures in this section assume that your projects have been prepared for internationalization. Preparation steps are in *Preparing a project to support internationalization, page 232*.

1. Store the translated data in a data warehouse. Translated data strings can be stored either in their own columns and/or tables in the same warehouse as the source (untranslated) data, or in different warehouses separated by language. Some organizations keep the source language stored in one warehouse, with all other languages stored together in a different warehouse. You must configure MicroStrategy with a DI model so it can connect to one of these storage scenarios: the SQL-based model and the connection-based model. For details on each model and steps to configure MicroStrategy, see *Storing translated data: data internationalization models, page 244*).
2. Enable the languages in MicroStrategy that will be supported by the project and configure the system based on where the translated data is stored (see *Enabling or disabling languages in the project to support DI, page 248*).

Storing translated data: data internationalization models

> This section assumes that you understand the structure of your organization's data storage. Table and column creation, maintenance, and alteration is beyond the scope of this guide. For information about data warehouses and how internationalization affects the process of storing and organizing information in the data warehouse, see the *MicroStrategy Project Design Guide*.

You must connect MicroStrategy to your storage system for translated data. To do this, you must identify which type of storage system you are using. Translated data for a given project is stored in one of two ways:

- In columns and tables within the same data warehouse as your source (untranslated) data (see *SQL-based DI model, page 245*)
- Stored in a different data warehouse from your source (untranslated) data (see *Connection-based DI model, page 245*)

SQL-based DI model

If all of your translations are stored in the same data warehouse as the source (untranslated) data, this is a SQL-based DI model. This model assumes that your translation storage is set up for column-level data translation (CLDT) and/or table-level data translation (TLDT), with standardized naming conventions.

This model is called SQL-based because SQL queries are used to directly access data in a single warehouse for all languages. You can provide translated DESC (description) forms for attributes with this DI model.

If you are using a SQL-based DI model, you must specify the column pattern or table pattern for each language. The pattern depends upon the table and column names that contain translated data in your warehouse. MicroStrategy supports a wide range of string patterns. The string pattern is not limited to suffixes only. However, using prefixes or other non-suffix naming conventions requires you to use some functions so that the system can recognize the location of translated data. These functions are included in the steps to connect the system to your database.

Regular (non-locale-specific) connection maps are treated normally by MicroStrategy if you choose the SQL-based DI model.

This model is recommended if using MicroStrategy Intelligent Cubes. For steps to point MicroStrategy to the correct columns or tables for each language, see *Connecting the system to a single database: SQL-based DI model, page 246*.

Connection-based DI model

If the translated data is stored in different data warehouses for each language, MicroStrategy retrieves the translations using a database connectivity API, namely ODBC. This model is called connection-based because a connection to more than one data warehouse must be made to access data in all languages. This is commonly called warehouse-level data translation (WLDT).

When using a connection-based DI model, you can connect to as many data warehouses as necessary, for example, one for each language. For steps to provide the appropriate database connection information for each data warehouse, see *Connecting the system to more than one database: connection-based DI model, page 247*.

Choosing a DI model

You must evaluate your physical data storage for both your source (untranslated) language and any translated languages, and decide which data internationalization model is appropriate for your environment.

MicroStrategy can use either a SQL-based or a connection-based DI model, but not both. For example, if your project supports 10 languages, and 5 of those languages are stored in one data warehouse and the other 5 are stored individually in separate data warehouses, MicroStrategy does not support this storage solution.

The following table describes common translation storage scenarios, and shows you which DI model and translation access method must be used.

Translation Storage Location	Translation Access Method	Data Internationalization Model
Different tables for each language, in one data warehouse	Different SQL generated for each language	SQL-based
Different columns for each language, in one data warehouse	Different SQL generated for each language	SQL-based
Different tables and columns for each language, in one data warehouse	Different SQL generated for each language	SQL-based
One data warehouse for each language	Different database connection for each language	Connection-based

If you are creating a new data warehouse and plan to implement DI, and you also use Intelligent Cubes, it is recommended that you use a SQL-based DI model, with different tables and/or columns for each language. Because a single Intelligent Cube cannot connect to more than one data warehouse, using a connection-based DI model requires a separate Intelligent Cube to be created for each language. This is very resource-intensive. For information about Intelligent Cubes in general and details on designing Intelligent Cubes for an internationalized environment, see the MicroStrategy *In-memory Analytics Guide*.

Connecting the system to the translation database

After languages have been enabled for the project, you must configure the system so that MicroStrategy can retrieve the translated data. This configuration varies depending on the data internationalization (DI) model used:

- **Connection-based DI model**: You must specify a database connection for each language.
- **SQL-based DI model**: You must specify a column pattern or table pattern for each language.

These models are described in detail in *Storing translated data: data internationalization models, page 244*.

Connecting the system to a single database: SQL-based DI model

For a detailed explanation of how to set up tables and columns to support SQL-based data internationalization, see the *Project Design Guide*, *Internationalization through tables and columns or databases* section. The *Project Design Guide* provides extensive examples and images of table and column naming patterns, explains the use of only tables, only columns, or both tables and columns, the use of logical views, and so on.

Your table suffixes for languages should be consistent and unified across the entire warehouse. For example, if you have Spanish translations in your warehouse, the suffix should be _SP for all tables that include Spanish translations, and not _SP, _ES, _EP, and so on.

For detailed steps to connect the system to your translation database, see the *Project Design Guide*, *Enabling data internationalization through SQL queries* section. The *Project Design Guide* includes details to select your table or column naming pattern, as well as functions to use if your naming pattern does not use suffixes.

If you are changing from one DI model to another, you must reload the project after completing the steps above. Settings from the old DI model are preserved, in case you need to change back.

Connecting the system to more than one database: connection-based DI model

If you are using a connection-based DI model, you must specify a database connection for each data warehouse that stores translated data.

Connection mapping can also be performed using Command Manager.

For a detailed explanation of how to set up your databases to support data internationalization, see the *Project Design Guide*, *Internationalization through tables and columns or databases* section. The Project Design Guide provides extensive examples and images of translation table structures in different databases, as well as important restrictions on logical views and supported character sets.

The database connection that you use for each data warehouse must be configured in MicroStrategy before you can provide translated data to MicroStrategy users.

Prerequisites

- The procedure in the Project Design Guide assumes that you will enable the connection-based DI model. If you decide to enable the SQL-based model, you can still perform the steps to enable the connection-based model, but the language-specific connection maps you create in the procedure will not be active.
- The physical schemas of all data warehouses to be used for data internationalization should be identical.
- You must have the Configure Connection Map privilege, at either the user level or the project level.
- Objects displayed in the Connection Mapping Editor are limited to those objects the user has Browse and Use permissions for.

For detailed steps to connect the system to more than one data warehouse, see the *Project Design Guide*, *Enabling data internationalization through connection mappings* section.

If you are changing from one DI model to another, you must reload the project after completing the steps in the P*roject Design Guide*. Settings from the old DI model are preserved, in case you need to change back.

You can delete a connection mapping by right-clicking on the connection map and selecting **Delete**. For details on general connection mapping procedures such as deletion, see the *MicroStrategy Developer Help* (formerly the *MicroStrategy Desktop Help*).

Supporting data internationalization for attribute elements

If you are using the SQL-based DI model, you must perform an additional step to support the display of translated attribute elements in reports and documents.

If the project designer has not already done so, you must define attribute forms in the project so that they can be displayed in multiple languages. Detailed information and steps to define attribute forms to support multiple languages are in the *Project Design Guide*, *Supporting data internationalization for attribute elements section*.

Enabling or disabling languages in the project to support DI

For languages that are stored in your data warehouse to be available for use in MicroStrategy, you must configure the project to support those languages.

You can also add a custom language to the list of languages available to be enabled for data internationalization. For steps to add a custom language to the project, see *Adding or removing a language in the system, page 270*.

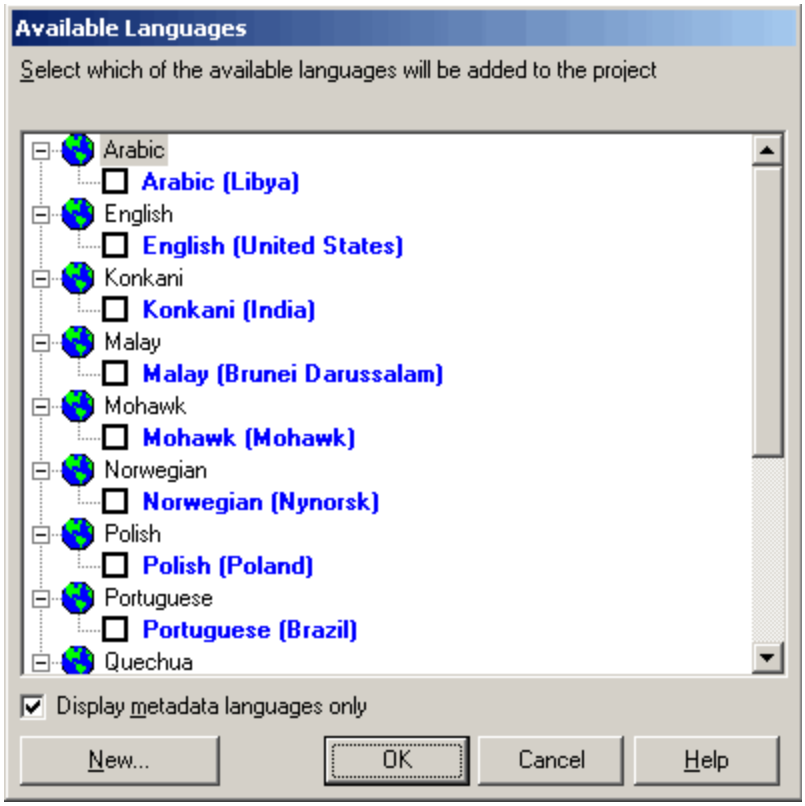
Enabling languages for data internationalization

After translated data has been stored, you must configure the project to establish which languages will be supported for data internationalization (DI). You must perform this procedure whether you store translated data using a SQL-based DI model or a connection-based DI model.

To enable data internationalization languages in a project

1. Log in to a project as a user with administrative privileges.
2. Right-click the project and select **Project Configuration**. The Project Configuration Editor opens.
3. On the left side of the Project Configuration Editor, expand **Language**, then select **Data**. The Language - Data dialog box opens on the right.
4. Select the **Enable data internationalization** check box.
5. Select the DI model that you are using. For details, see *Storing translated data: data internationalization models, page 244*.
   - For a SQL-based DI model, select **SQL based**.
   - For a connection-based DI model, select **Connection mapping based**.

6 Click **Add.** The Available Languages dialog box opens, as shown below:



7 Languages displayed in bold blue are those languages that have been enabled for the project to support translated metadata objects, if any. This list is displayed as a starting point for the set of languages you can choose to enable for supporting data internationalization.

- To display all available languages, or if no metadata languages are displayed, clear the **Display metadata languages only** check box.
- To add a new language, make sure the **Display metadata languages only** check box is cleared, and then click **New**. The Languages Editor opens. For steps to create a custom language, see ***Adding or removing a language in the system, page 270***.

8 Select the check box next to any language or languages that you want to enable for this project.

If no languages are selected to be enabled to support data internationalization, then data internationalization is treated by the system as disabled.

9 Click **OK.** The Available Languages dialog box closes.

- The Change Comments dialog box opens. You can type any internal comments to provide other administrators with information about the steps you are taking to support data internationalization.

The languages you selected are displayed in the Language: Data dialog box.

10 In the **Default** column, select one language to be the default language for data internationalization in the project. This selection does not have any impact on the project or how languages are supported for data internationalization. Unlike the MDI default language, this DI default language can be changed at any time.

If no default DI language is selected, data internationalization is treated by the system as disabled.

11 For each language you have enabled, define the column/table naming pattern or the connection-mapped warehouse, depending on which DI model you are using (for information on DI models and on naming patterns, see *Storing translated data: data internationalization models, page 244*):

- **SQL-based DI model**: If you selected the SQL-based DI model above, click the **Column Pattern** and **Table Pattern** columns next to one of the languages you will support. Type the column or table prefix or suffix and click **OK**. For examples, click **Help**.
- **Connection-based DI model**: If you selected the connection-based DI model above, click the **Database Connection** column next to one of the languages you will support. Select the appropriate warehouse from the drop-down list.

Some languages may have the same suffix - for example, English US and English UK. You can also specify a NULL suffix.

12 Click **OK**. The Project Configuration Editor closes.

13 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Disabling a language for data internationalization

You can use the steps below to disable a language for a project. When a language has been disabled in a project, that language is no longer available for users to select as a language preference, and the language cannot be seen in any translation-related interfaces such as an object's default language in its Properties - International dialog box. Any translations for the disabled language are not removed from the data warehouse with these steps.

If a user has selected the language as a language preference, the preference will no longer be in effect once the language is disabled. The project's default language will take effect.

If you remove the language currently set as the default data internationalization language, the system automatically selects the first language in the list of remaining enabled languages to set the new default language. This new default data internationalization language should not have any impact on your project.

If you disable all languages for data internationalization (DI), the system treats DI as disabled. Likewise, if you do not have a default language set for DI, the system treats DI as disabled.

To disable data internationalization languages in a project

1. Log in to a project as a user with administrative privileges.
2. Right-click the project and select **Project Configuration.** The Project Configuration Editor opens.
3. On the left side of the Project Configuration Editor, expand **Language**, then select **Data**.
4. On the right side, under Selected Languages, clear the check box for the language that you want to disable for the project.
5. Click **OK.** The Project Configuration Editor closes.
6. Perform the following steps depending on how your project is affected:
   - Empty any caches or Intelligent Cubes containing content in the disabled DI language.
   - Language disabling will only affect MDX cubes and regular reports and documents if an attribute form description in the disabled language exists in the cube or report. If this is true, the cube, report, or document cannot be published or used. The cube, report, or document designer must remove attribute forms in the disabled language before the cube/report/document can be used again.
7. Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Making translated data available to users

After you have performed the necessary steps to configure metadata object translation and/or data translation in the system, you can specify which language(s) should be displayed for various users in the interface and in reports (both report objects and report results). You can specify language preferences at the project level and at the all-projects level. By selecting various levels of language preferences, you specify which language is preferred as a fallback if a first choice language is not available.

These language preferences are for metadata languages only. All data internationalization languages fall back to the project's default language if a DI preference is not enabled or translation of a specific report cell is not available.

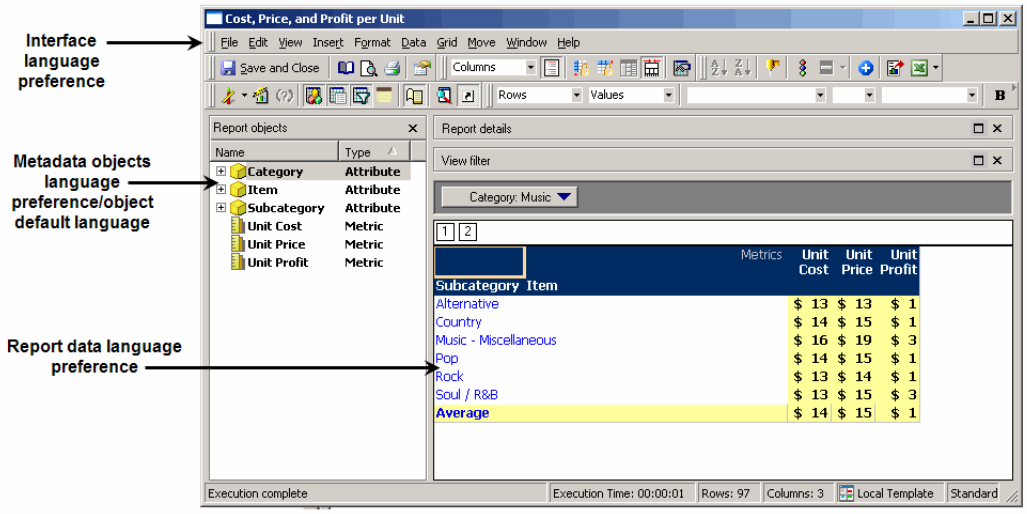
The following sections show you how to select language preferences based on various priority levels within the system, starting with a section that explains the priority levels:

- *Selecting preferred languages for interfaces, reports, and objects, page 252*
- *Selecting the Interface language preference, page 253*
- *Configuring metadata object and report data language preferences, page 254*
- *Selecting the object default language preference, page 264*

Selecting preferred languages for interfaces, reports, and objects

After translated data is stored in your data warehouse and/or metadata database, and languages have been enabled for the project, you must specify which languages are the preferred languages for the project and the user. These selected languages are called language preferences.

The following image shows the different parts of the MicroStrategy environment that display translated strings based on the language preferences:



The following language preferences can be configured:

- **Interface Language**: Determine the language that menu options, dialog box text, and so on, will display. For steps to set this preference, see *Selecting the Interface language preference, page 253*.
- **Metadata objects:** Determine the language that will be displayed for MicroStrategy objects that come from the metadata database, such as metric names, report names, system folder names, and so on. For steps to set this preference, see *Configuring metadata object and report data language preferences, page 254*.
- **Report data**: Determine the language that will be displayed for report results that come from your data warehouse, such as attribute element names. For steps to set this preference, see *Configuring metadata object and report data language preferences, page 254*.
- **Object default language**: Determine the fallback language for MicroStrategy objects. This language is used if a report is executed in a language that the object lacks a translation for. For steps to set or change this default preference, see *Selecting the object default language preference, page 264*.

Each language preference can be configured independently of the others. For example, it is possible to have a report that displays all metadata object names in French, while any data from the data warehouse is displayed in English, and the interface is translated into Spanish. However, for best performance it is recommended that you use a unified

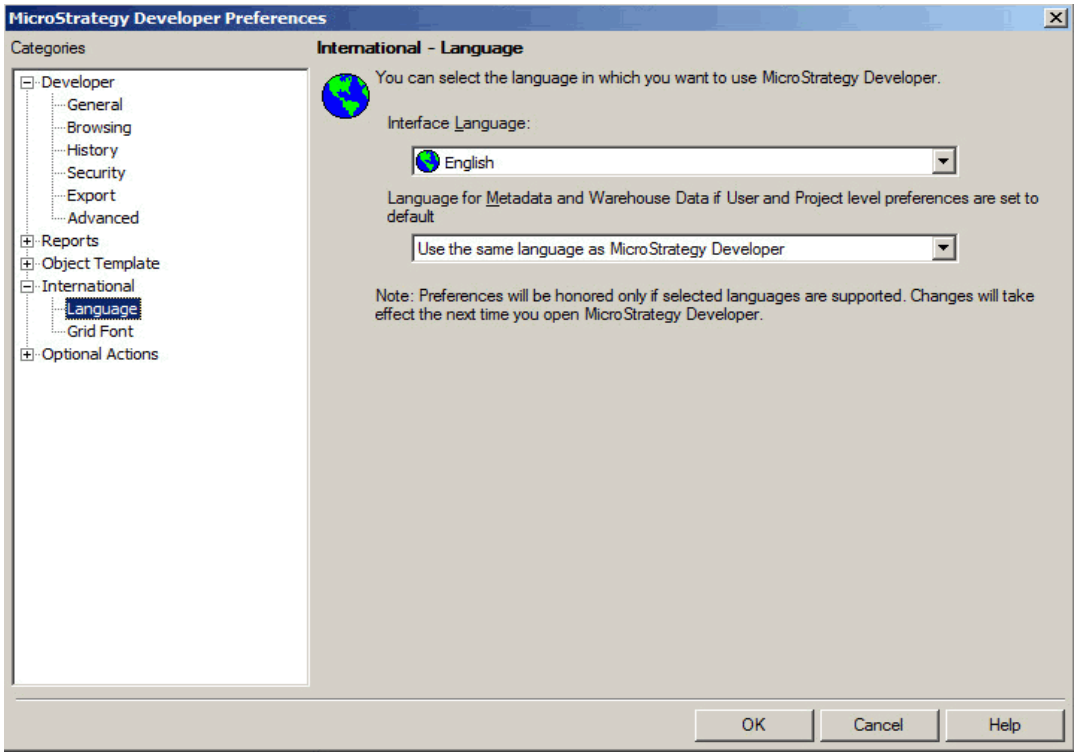
language display in Developer. For example, if you use French for the interface, the metadata objects language preference and the report data language preference, as well as number and date preferences, should also be in French.

Selecting the Interface language preference

The interface language preference determines what language Developer menus, editors, dialog boxes, monitors and managers, and other parts of the Developer software are displayed in. Use the steps below to set this preference.

Configuring the Interface language preference

1. In Developer, log in to the project.
2. From the **Tools** menu, select **Preferences**. The Developer Preferences dialog box opens.
3. On the left, expand **International** and select **Language.** The International: Language dialog box is displayed, as shown below:



4. From the **Interface Language** drop-down list, select the language that you want to use as the interface default language

The interface language preference can also be used to determine the language used for the metadata objects and report data, if the Developer level language preference is set to **Use the same language as MicroStrategy Developer.** For more information on the Developer level language preference, see *Selecting the Developer level language preference, page 260*.

5 Select **OK.** The Developer Preferences dialog box closes.

6 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Configuring metadata object and report data language preferences

There are several levels at which metadata and report data languages can be specified in MicroStrategy. Lower level languages are used by the system automatically if a higher level language is unavailable. This ensures that end users see an appropriate language in all situations.

Language preferences can be set at six different levels, from highest priority to lowest. The language that is set at the highest level is the language that is always displayed, if it is available. If that language does not exist or is not available in the metadata or the data warehouse, the next highest level language preference is used.

If a language preference is not specified, or is set to Default, MicroStrategy automatically uses the next lower priority language preference. If none of these language preferences are set, the interface preferred language is used.

When an object is created, its default object language is automatically set to match the creator's metadata language preference. If the creator has his metadata language preference set to Default, the new object's default language is decided based on the rules in this section: the system will first try to use a default language for all users of the project, then a language preference set for all users of Developer, then the default language set for the project (as shown in the table below).

The following table describes each level, from highest priority to lowest priority, and points to information on how to set the language preference at each level.

- End user preference settings override any administrator preference settings, if the two settings conflict.
- Distribution Services deliveries are one exception to the hierarchy below. For details, see *Selecting the Machine level language preference, page 262*.

Language Preference Level (highest to lowest priority)	Description	Setting Location for End Users	Setting Location for Administrators
User-Project level	The language preference for a	Web: **Preferences** link at the top of any page.	Set in the User Language Preference Manager. See

Language Preference Level (highest to lowest priority)	Description	Setting Location for End Users	Setting Location for Administrators
	user for a specific project.	Developer: From the **Tools** menu, select **My Preferences**.	*Selecting the User-Project level language preference, page 256*.
User-All Projects level	The language preference for a user for all projects.	Web: **Preferences** link at the top of any page. Developer: From the **Tools** menu, select **My Preferences**.	Set in the User Editor. See *Selecting the User-All Projects level language preference, page 258*.
Project-All Users level	The language preference for all users in a specific project.	Not applicable.	In the Project Configuration Editor, expand **Languages**, select **User Preferences**. See *Selecting the All Users In Project level language preference, page 259*.
Developer level	The interface language preference for all users of Developer on that machine, for all projects.	Set in the Developer Preferences dialog box. For steps to specify this language, see *Selecting the Developer level language preference, page 260*.	Set in the Developer Preferences dialog box. For steps to specify this language, see *Selecting the Developer level language preference, page 260*.
Machine level	The language preference for all users on a given machine.	On the user's machine and within the user's browser settings.	On the user's machine and within the user's browser settings. For steps to specify this language, see *Selecting the Machine level language preference, page 262*.
Project Default level	This is the project default language set for MDI. It is the language preference for all users connected to the metadata.	Not applicable.	Set in the Project Configuration Editor. For steps to specify this language, see *Configuring the Project Default level language preference, page 262*.

For example, a user has her User-Project Level preference for Project A set to English. Her User-All Projects Level preference is set to French. If the user logs in to Project A and runs a report, the language displayed will be English. If the user logs in to Project B, which does not have a User-Project Level preference specified, and runs a report, the project will be displayed in French. This is because there is no User-Project Level preference for Project B, so the system automatically uses the next, lower language preference level (User-All Projects) to determine the language to display.

These language preferences apply to strings translated in both the metadata and the data warehouse. However, MicroStrategy handles missing translations differently, depending upon whether the string is translated in the metadata or the data warehouse:

- **Metadata**: When a translation for an object in the metadata is missing in the preferred language, the object default language preference is used. For more information about the object default language preference, see *Selecting the object default language preference, page 264*.
- **Data warehouse**: When a translation for data in the data warehouse is missing in the preferred language (the column or table is present in the data warehouse but is empty), the report returns no data.

The following sections provide steps to configure each preference level, starting from the highest priority and ending at the lowest priority.

Selecting the User-Project level language preference

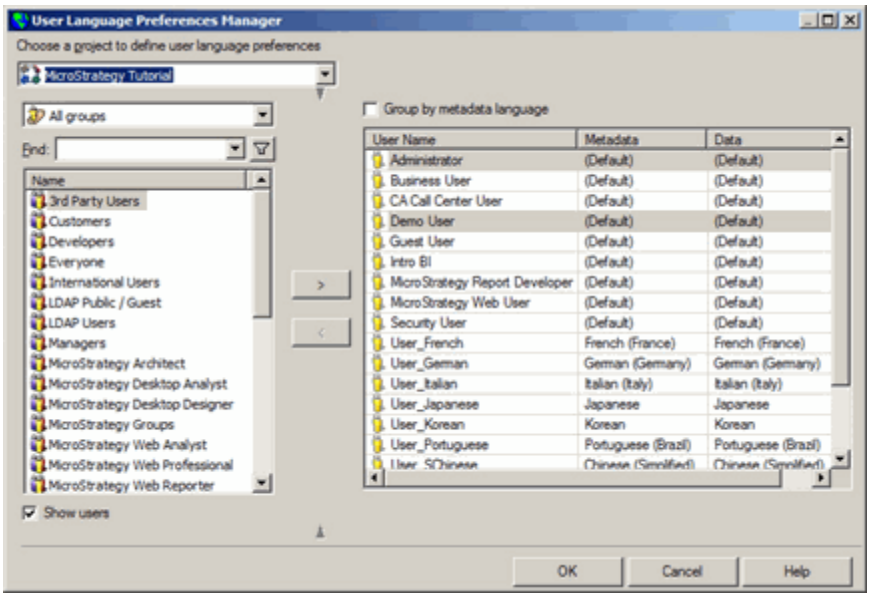
The User-Project Level language preference is the language preference for a given user for a specified project. It is the highest priority language setting; to see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 254*.

This preference is specified in the User Language Preference Manager in Developer. Use the steps below to set this preference.

If an object has an empty translation in a user's chosen project language preference, the system defaults to displaying the object's default language, so it is not necessary to add translations for objects that are not intended to be translated.

Selecting the User-Project level language preference

1. Log in to Developer as a user with Administrative privileges.
2. Right-click the project that you want to set the language preference for and select **Project Configuration**. The Project Configuration Editor opens.
3. On the left side of the Project Configuration Editor**,** expand **Languages**, and select **User Preferences**.
4. On the right side, under **User Language Preference Manager**, click **Modify**. The User Language Preference Manager opens, shown below:



5 In the **Choose a project to define user language preferences** drop down menu at the top left, select the appropriate project.

6 Select the users from the list on the left side of the User Language Preferences Manager that you want to change the User-Project level language preference for, and click > to add them to the list on the right. You can narrow the list of users displayed on the left by doing one of the following:

- To search for users in a specific user group, select the group from the drop-down menu that is under the **Choose a project to define user language preferences** drop-down menu.
- To search for users containing a certain text string, type the text string in the **Find** field, and click the following icon:

This returns a list of users matching the text string you typed.

Previous strings you have typed into the **Find** field can be accessed again by expanding the **Find** drop-down menu.

7 On the right side, select the user(s) that you want to change the User-Project level preferred language for, and do the following:

You can select more than one user by holding CTRL.

- Select the desired language to be applied to translated metadata objects from the drop-down menu in the **Metadata** column. This language will be displayed for the selected user(s) when connecting to the selected project.
- Select the desired language to be applied to report results from the drop-down menu in the **Data** column. This language will be displayed for the selected user (s) when connecting to the selected project.

8 Click **OK.** The preferences are saved and the User Language Preferences Manager closes.

Once the user language preferences have been saved, users can no longer be removed from the **Selected** list.

9 Click **OK.** The Project Configuration Editor closes.

10 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the User-All Projects level language preference

The User-All Projects level language preference determines what language will be applied to all projects that a specific user sees when connected to a project source, unless a higher priority language preference has been specified for the user. Use the steps below to set this preference.

If the User-Project language preference is specified for the user, the user will see the User-All Projects language only if the User-Project language is not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 254*.

Selecting the User-All Projects level language preference

1 Log in to Developer as a user with Administrative privileges.

2 In the Folder List on the left, within the appropriate project source, expand **Administration,** expand **User Manager**, and navigate to the user that you want to set the language preference for.

3 Double-click the user. The User Editor opens.

4 On the left side of the User Editor, expand the **International** category and select **Language.**

5 On the right side of the User Editor, do the following, depending on whether you have configured metadata object translation (MDI) or data warehouse translation (DI), or both:

- Select the language that you want to be applied to translated metadata strings from the **Default metadata language preference for this user** drop-down menu.
- Select the language that you want to be applied to translated data warehouse strings from the **Default data language preference for this user** drop-down menu.

6 Click **OK.** The User Editor closes.

7 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the All Users In Project level language preference

The All Users In Project level language preference determines the language that will be displayed for all users that connect to a project, unless a higher priority language is specified for the user. Use the steps below to set this preference.

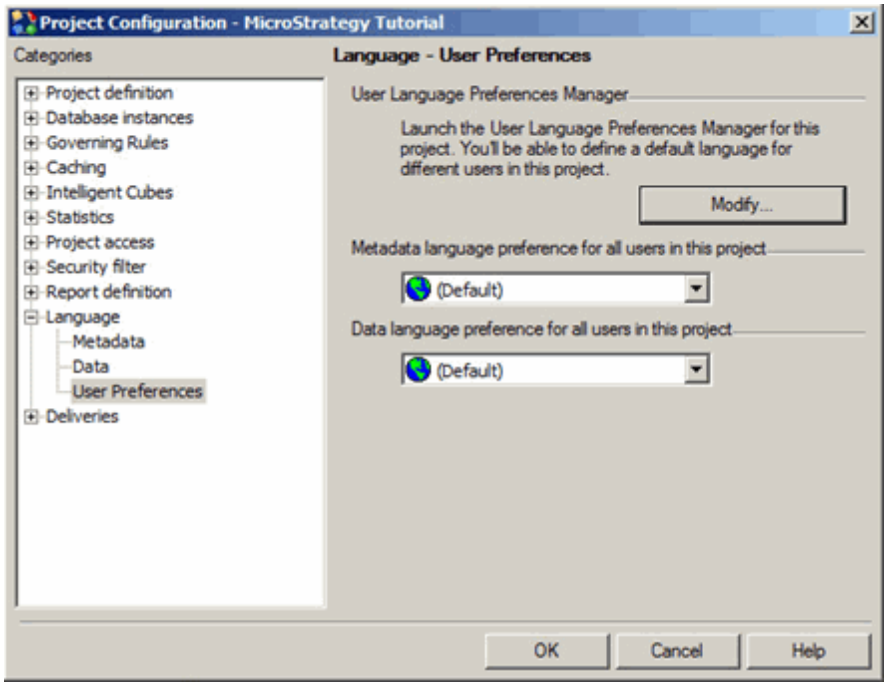
If the User-Project or User-All Projects language preferences are specified for the user, the user will see the All Users In Project language only if the other two language preferences are not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 254*.

Selecting the All Users In Project level language preference

1 Log in to Developer as a user with Administrative privileges.

2 In the Folder List on the left, select the project. From the **Administration** menu, select **Projects**, then **Project Configuration.** The Project Configuration Editor opens.

3 On the left side of the Project Configuration Editor, expand **Languages** and select **User Preferences**. The Language-User Preferences dialog box is displayed, as shown below:



4 Do the following, depending on whether you have configured metadata object translation (MDI) or data warehouse translation (DI), or both:

- From the **Metadata language preference for all users in this project** drop-down menu, select the language that you want to be displayed for metadata object names in this project.
- From the **Data language preference for all users in this project** drop-down menu, select the language that you want to be displayed for report results in this project.

5 Click **OK.** The Project Configuration Editor closes.

6 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the Developer level language preference

The Developer level language preference determines the default language for all objects displayed within Developer, unless a higher priority language preference has been specified. This is the same as the interface preference.

If the User-Project, User-All Projects, or All Users In Project language preferences are specified, the user will see the Developer language only if the other three language preferences are not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 254*.

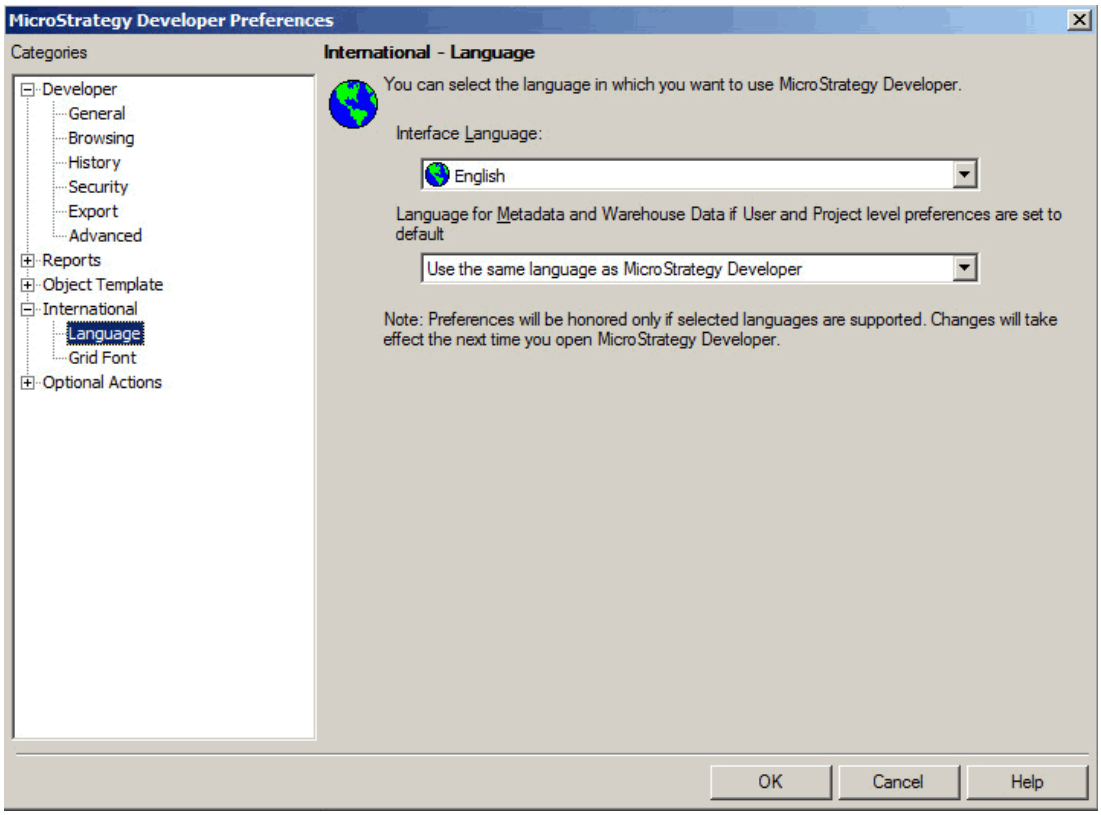
This language preference must be configured to match one of two other language preferences: the Interface language preference or the Machine level language preference. For information about the Interface language preference, see *Selecting the Interface language preference, page 253*. For information about the Machine level language preference, see *Selecting the Machine level language preference, page 262*

Selecting the Developer level language preference

1 Log in to Developer as a user with Administrative privileges.

2 From the **Tools** menu, select **MicroStrategy Developer Preferences.** The Developer Preferences dialog box opens.

3 Expand the **International** category and select **Language.** The International - Language dialog box opens, as shown below:



4 Select one of the following from the **Language for metadata and warehouse data if user and project level preferences are set to default** drop-down menu.

- If you want the Developer language preference to be the same as the Interface language preference, select **Use the same language as MicroStrategy Developer.** For information about configuring the Interface language preference, see ***Selecting the Interface language preference, page 253***.
- If you want the Developer language preference to be the same as the Machine-level language preference, select **Use language from Regional Settings**. For information about configuring the Machine-level language preference, see ***Selecting the Machine level language preference, page 262***.

5 Select the language that you want to use as the default Developer interface language from the **Interface Language** drop-down menu.

6 Click **OK.** The Developer Preferences dialog box closes.

7 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the Machine level language preference

This preference determines the language that is used on all objects on the local machine. MicroStrategy Web uses the language that is specified in the user's web browser if a language is not specified at a level higher than this one.

- If the User-Project, User-All Projects, All Users In Project, or Developer language preferences are specified, the user will see the Machine language only if the other four language preferences are not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 254*.
- A MicroStrategy Distribution Services delivery (such as an email, file, or printer delivery) uses a different language resolution logic: If the User-Project, User-All Projects, All Users in Project, and Developer languages are not able to be displayed, the delivery defaults to the Project Default level language preference, followed by the Machine level language preference. This is because Distribution Services runs without a client session in the Intelligence Server machine; if the Machine level language took precedence, all users receiving delivered content would receive that content using the Intelligence Server machine's language. Instead, the project's default language is the fallback language for Distribution Services deliveries.

To select the Machine level language preference on a Windows machine, from the **Start** menu, select **Control Panel**, then **Regional and Language Options**. Consult your machine's Help for details on using the language options.

Configuring the Project Default level language preference

This language preference specifies the default language for the project. This language preference has the lowest priority in determining the language display. Use the steps below to set this preference.

- If the User-Project, User-All Projects, All Users In Project, Developer, or Machine-level language preferences are specified, the user will see the Project Default language only if the other five language preferences are not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 254*.
- A MicroStrategy Distribution Services delivery (such as an email, file, or printer delivery) uses a different language resolution logic: If the User-Project, User-All Projects, All Users in Project, and Developer languages are not able to be displayed, the delivery defaults to the Project Default level language preference, followed by the Machine level language preference. This is because Distribution Services runs without a client session in the Intelligence Server machine; if the Machine level language took precedence, all users receiving delivered content would receive that content using the Intelligence Server machine's language. Instead, the project's default language is the fallback language for Distribution Services deliveries.

Selecting the Project Default language preference

The project default language is selected either when a project is first created, or the first time metadata languages are enabled for the project. It cannot be changed after that point. The following steps assume the project default language has not yet been selected.

1. Log in to the project as a user with Administrative privileges.
2. Select the project that you want to set the default preferred language for.
3. From the **Administration** menu, select **Project,** then **Project Configuration.** The Project Configuration Editor opens.
4. On the left side of the Project Configuration Editor, expand **Language**. Do one or both of the following, depending on whether you have configured metadata object translation (MDI) or data warehouse translation (DI), or both:
   - To specify the default metadata language for the project, select **Metadata** from the **Language** category. Then select **Default** for the desired language.
   - To specify the default data language for the project, select **Data** from the **Language** category. Then select **Default** for the desired language.
5. Select **OK.** The Project Configuration Editor closes.
6. Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the object default language preference

Each MicroStrategy object can have its own default language. The translation for the object default language is used when the system cannot find or access a translation for the object in the language specified as the user or project preference.

This preference is useful especially for personal objects, since most personal objects are only used in one language, the owner's language. The object default language can be set to any language supported by the project in which the object resides.

Some objects may not have their object default language preference set, for example, if objects are merged from an older, non-internationalized MicroStrategy system into an upgraded, fully internationalized environment. In this case, for those objects that do not have a default language, the system automatically assigns them the project's default language.

This is not true for newly created objects within an internationalized environment. Newly created objects are automatically assigned the creator's metadata language preference. For details on the metadata language, see *Configuring metadata object and report data language preferences, page 254*.

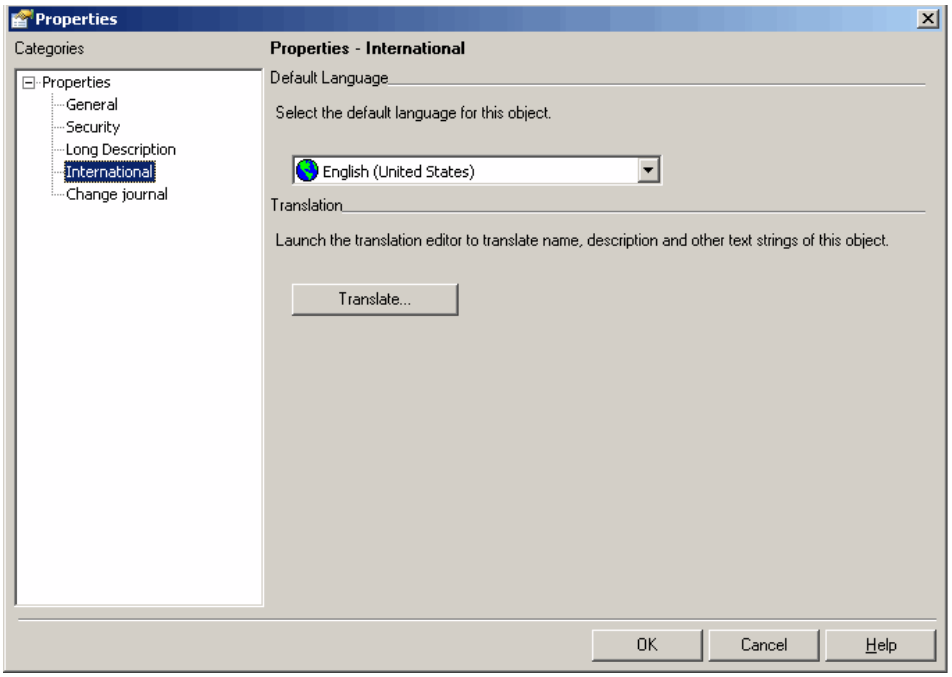
When duplicating a project, objects in the source that are set to take the project default language will take whatever the destination project's default language is.

Use the steps below to configure the object default language.

For the hierarchy of language preferences, see the table in *Configuring metadata object and report data language preferences, page 254*.

Configuring the object default language preference

1. Log in to the project source that contains the object as a user with administrative privileges.
2. Right-click the object and select **Properties.** The Properties dialog box opens.
   - You can set the default language for multiple objects by holding the **Ctrl** key while selecting multiple objects.
3. Select **International.** The Properties - International dialog box is displayed, as shown below:



If the International option is missing, the object is not supported for translation. For example, there is no reason to translate a table name for a schema object (such as LU_YEAR), so this object does not have the International option available.

4 From the **Select the default language for the object** drop-down menu, select the default language for the object(s).

5 Click **OK.** The Properties dialog box closes.

Achieving the correct language display

The following table lists many of the locations where you might want to display a given language for users. It tells you where to configure the system so that the language is displayed or available for selection. For some language displays, there are different steps in Developer than in MicroStrategy Web.

Translation or Language Display that You Want to Achieve	Where to Enable It
Number format (decimal, thousands separator, currency symbol, weight)	In Developer: Regional settings on the Developer user's machine. In Web: Click the **MicroStrategy** icon, then select **Preferences**. You can create a dynamic currency format that changes according to the locale's default currency symbol. The dynamic format applies to grid reports, graph reports, and documents displayed in MicroStrategy Web, MicroStrategy

Translation or Language Display that You Want to Achieve	Where to Enable It
	Mobile, and MicroStrategy Office and exported to PDF. For a graph report, the dynamic currency is applied to the data label. Click **Help** for steps and details on creating the dynamic format.
Currency conversion	Use a Value prompt on a metric. See the *Advanced Prompts* chapter of the Advanced Reporting Guide.
Date format and separators	In Developer: Regional settings on the Developer user's machine. In Web: Click the **MicroStrategy** icon, then select **Preferences**. Select **Languages**, then select **Show Advanced Options**. **Note**: In Web, if the browser is set to a language unsupported in MicroStrategy and the user's preferences are set to Default, the date/time and number formatting display in English.
Autostyle fonts that support a given language	In Developer, right-click and **Format** the attribute or metric (column header, value, or subtotal) using the font you prefer (on the **Font** tab, specify the font.) From the **Grid** menu, select **Save Autostyle As** and either overwrite the existing autostyle or create a new one.
Fonts that support all languages	Few fonts support all languages. One that does is Arial Unicode MS, which is licensed from Microsoft.
PDFs, portable PDFs, bookmarks in PDFs, and language display in a Report Services document	Embed fonts when you are designing the document; this ensures that the fonts selected by the document designer are used to display and print the PDF, even on machines that do not have the fonts installed. Embedding fonts lets you: • Use language fonts other than Simplified/Traditional Chinese, English, Japanese, Korean, and Western European in PDFs. • Provide a true unicode environment, where one document can contain different languages. • Create portable PDFs to email and to publish in Web. To embed fonts, in the Document Editor in Developer, from the **Format** menu, select **Document Properties**, click **Export**, and select **Embed fonts in PDF**. To view embedded fonts in Developer, the fonts must be installed on the Developer machine and the Intelligence Server machine. To view embedded fonts in Web, the fonts must be installed on the Intelligence Server machine. To display PDF bookmarks with the correct font, the language pack must be installed on the viewer's machine. This is true for any language other than English or Western European.

Translation or Language Display that You Want to Achieve	Where to Enable It
Character sets in Teradata databases	The **Character Column Option** and **National Character Column Option** VLDB properties let you support the character sets used in Teradata. For examples and details to enable these properties, see *Chapter 1, SQL Generation and Data Processing: VLDB Properties*.
Double-byte language support	In Developer, from the **Tools** menu, select **Developer Preferences**.
User changing own language	In Web: Click the **MicroStrategy** icon, then select **Preferences**. In Developer: From the **Tools** menu select **Developer Preferences**. The list of languages to choose from comes from the languages enabled for a project; see *Enabling metadata languages for an existing project, page 235*.
Default language preference for a particular user	In the User Editor, expand **International**, and then select **Language**. An administrator needs the Use User Editor and Configure Language Settings privileges, and ACL permissions to modify the user object.
Default language for all users in a project	Right-click a project, select **Project Configuration**, expand **Language**, and select **User Preferences**.
Different default language for a single user in different projects	Right-click a project, select **Project Configuration**, expand **Language**, and select **User Preferences**.
Translating the project's default language	By default, the project's default language cannot be translated in the Object Translation Editor. The first column in the editor corresponds to the project's default language. To translate terms in the default language, in the Object Translation Editor, click **Options** at the top of the Editor. Move the default language from the Selected View Languages box to the Selected Edit Languages box.
Function names	Function names are not translated. The MicroStrategy system expects function names to be in English.
An individual object	Use the Object Translation Editor. To access this, right-click the object and select Translate.
Caches in an internationalized environment	See *Caching and internationalization, page 230*.
Intelligent Cubes	It is recommended that you use a SQL-based DI model when setting up internationalization, as described in *Providing data internationalization, page 244*. Because a single Intelligent Cube cannot connect to more than one data warehouse, using a connection-based DI model requires a separate Intelligent Cube to be created for each language, which can be resource-

Translation or Language Display that You Want to Achieve	Where to Enable It
	intensive. Details on this cost-benefit analysis, steps to enable a language when publishing an Intelligent Cube, and background information on Intelligent Cubes is in the *MicroStrategy In-memory Analytics Guide*.
Subscriptions in an internationalized environment	Subscribed-to reports and documents behave like standard reports and documents, and are delivered in the language selected in My Preferences or User Preferences.
Repository Translation Wizard list of available languages	Enable languages the project supports for metadata objects (see *Enabling metadata languages for an existing project, page 235*).
Metadata object names and descriptions (such as report names, metric names, system folder names, embedded descriptors such as attribute aliases, prompt, instructions, and so on)	For a new project being created, select these in Architect. You can view the database table columns used for internationalization as you create the project. For an existing project, see *Enabling metadata languages for an existing project, page 235*.
Configuration objects in Developer	Displayed according to User-Project level language preference. Set this by right-clicking the project, selecting **My Preferences**, selecting **International**, and setting the Metadata language for All Projects.
Attribute elements, for example, the Product attribute has an element called DVD player	First translate the element name in your data warehouse. Then enable the language; see *Enabling languages for data internationalization, page 248*.
Project name and description	In the Project Configuration Editor, expand **Project Definition**, select **General**, click **Modify**, select **International**, then click **Translate**. You can type both a project name and a description in the Object Description field.
When designing a project using Architect, see columns in the Warehouse Tables area that support data internationalization	In Architect, from the **Options** menu, select **Settings**. On the **Display Settings** tab, select **Display columns used for data internationalization**.
Enable a new language for a project to support	See *Enabling metadata languages for an existing project, page 235*. User adding the language must have Browse permission for that language object's ACL.
Enable a custom language for a project to support	See *Adding a new language to the system, page 270*. Then see *Enabling metadata languages for an existing project, page 235*. User adding the language must have Browse permission for that language object's ACL.
Searching the project	Searches are conducted in the user's preferred metadata language by default.

Translation or Language Display that You Want to Achieve	Where to Enable It
	A language-specific search can be conducted; open a project, then from the **Tools** menu select **Search for Objects**. On the International tab, click **Help** for details on each option in the interface.
Project or object migration, or duplication	Object Manager, Project Merge, and the Project Duplication Wizard contain translation-specific conflict resolution options for migrating translated objects between projects. Click **Help** in any of these tools for details, or see the *Managing Your Projects* chapter in the System Administration Guide.
Derived elements	In the Derived Element Editor, from the **File** menu, select **Properties**, then select **International**. For details to use the options, click **Help**.
MicroStrategy Office user interface and Excel format languages	In MicroStrategy Office, select **Options**, select **General**, then select **International**.
MDX (Multidimensional Expressions) data sources	MicroStrategy passes a user's required language as a database connection parameter to the MDX cube provider; the cube provider supplies the correct translations.

Maintaining your internationalized environment

You can add or remove languages from your MicroStrategy system, and you can edit the language objects in the system. You can use MicroStrategy Command Manager to automate several maintenance tasks. MicroStrategy Object Manager and Project Merge contain some translation-specific options for conflict resolution rules.

These maintenance processes and tools are described below. This section also covers security and specialized translator user roles.

Using Command Manager to automate language maintenance tasks

Several Command Manager scripts are designed to make language maintenance and user maintenance related to internationalized environments easier and faster. These scripts include:

- List all languages (metadata or data) by project, or all languages contained under Administration > Configuration Managers > Languages in Developer's Folder List.
- List available languages (metadata or data) at a specified level, such as by user/project, or by user and project.
- List resolved languages, which are the languages that are displayed to users from among the list of possible preferences.
- Alter languages at a specified level, which changes language preferences for a set of users or for a project.

For these and all the other scripts you can use in Command Manager, open Command Manager and click **Help**. For background information on Command Manager, see the *Command Manager* chapter in the *MicroStrategy System Administration Guide*.

Moving translated objects between projects

You can use Object Manager and Project Merge to migrate translated objects between projects. You apply the same MicroStrategy conflict resolution rules as you use when merging non-translated objects, but you use these rules specifically for the translated names and descriptions that are part of each translated object. You can also merge translations even if objects are identical. For details on all the options for migrating translated objects using Object Manager or Project Merge, open Object Manager or Project Merge and click **Help**.

Adding or removing a language in the system

You can add or remove languages and language variants from the system using the steps below.

Supporting character sets

Languages require a wide range of character sets to represent data. To support the languages you plan to use in your MicroStrategy projects, you must use databases that support the required character sets and that are configured accordingly. To determine whether your database supports the character sets required to display the languages you want to support, see your third-party database documentation.

Specifically, the database that allocates the metadata must be set with a code page that supports the languages that are intended to be used in your MicroStrategy project.

Adding a new language to the system

You can add new languages to MicroStrategy. Once they are added, new languages are then available to be enabled for a project to support internationalization.

Variant languages (also called custom languages) can also be added. For example, you can create a new language called Accounting, based on the English language, for all users in your Accounting department. The language contains its own work-specific terminology.

Prerequisites

- You must have the Browse permission for the language object's ACL (access control list). For details on ACLs, see the *System Administration Guide*.

To add a new language to the system

1 Log in to a project as a user with administrative privileges.

2 Right-click the project and select **Project Configuration.** The Project Configuration Editor opens.

3 On the left side of the Project Configuration Editor, expand **Language**, then select either **Metadata** or **Data**, depending on whether you want to add the language to support metadata objects or to support data internationalization. For a description of the differences, see *About internationalization, page 229*.

4 Click **Add.** The Available Languages dialog box opens.

5 Click **New**. The Languages Editor opens. Click **Help** for details on each option.

6 Click **OK**. The Project Configuration Editor closes.

7 If the language you added to the system is certified by MicroStrategy, you are prompted to automatically update system object translations that come with MicroStrategy. The information that is automatically updated includes translations of the following:

- System folders: The Public Objects folder and the Schema Objects folder
- Project objects: Autostyles and object templates
- System configuration objects: Security roles and user groups

Click **Yes**. You can also perform this update later, using the Project Configuration Editor, and selecting **Upgrade** in the Project Definition category.

8 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Languages can also be added using the Languages Configuration Manager, by expanding **Administration** within the appropriate project source in the Folder List, expanding **Configuration Managers**, and selecting **Language**.

After adding a new language, if you use translator roles, be sure to create a new user group for translators of the new language (see *Creating translator roles, page 274*).

To add a new interface language for MicroStrategy Web users

This procedure provides high-level steps for adding a new language to the display of languages in MicroStrategy Web. After the new language is added, Web users can select this language for displaying various aspects of Web in the new language. For details and best practices to customize your MicroStrategy Web files, see the MicroStrategy Developer Library (MSDL), which is part of the MicroStrategy SDK.

1 In the locales.xml file (located by default in <application-root-path>/WEB-INF/xml), add a new line for the language key using the example below:

<locale locale-id=" 13313" language="HI" country="Ind" desc="Hindi" desc-id="mstrWeb.5097" char-set="UTF-8" char-set-excel="UnicodeLittle" codepage="65001" codepage-excel-"1252"/>

2. Create resource files for the new language, for generic descriptors, based on existing resource files. For example:
   - For Web messages: Messages_Bundle_HI.properties
   - For number and date formats in the interface: Format_Config_HI.xml
3. If you want to display feature-specific descriptors for the new language, you can create resource files based on existing resource files. For example:
   - DashboardDatesBundle_13313.xml
   - DashboardViewerBundle_13313.xml

Creating a language variant: Multi-tenancy

A language variant is a language that is similar to a standard language, because the variant is based on the standard language. A variant can be created for a specific purpose in an organization, for example, Executive Business English.

Multi-tenancy is providing numerous groups of users access to the same MicroStrategy environment, but changing the display of objects and object names or descriptions based on various configuration settings. For more information on multi-tenancy, see *Multi-Tenant Environments: Object Name Personalization, page 280*.

Removing a language from the system

A language cannot be removed from the system if it is being used by a project, that is, if it has been enabled to be supported for a project. To remove a language from a project, that language must first be disabled from the project, as described in the steps below.

If a user has selected the language as a language preference, the preference will no longer be in effect once the language is disabled. The next lower priority language preference will take effect. To see the language preference priority hierarchy, see *Configuring metadata object and report data language preferences, page 254*.

To remove a language from the system

1. Disable the language from all projects in which it was enabled:
   - To disable a metadata language from a project, see *Enabling and disabling metadata languages, page 235*.
   - To disable a data language from the project, see *Enabling languages for data internationalization, page 248*.
2. For metadata languages, any translations for the disabled language are not removed from the metadata with these steps. To remove translations:
   - For individual objects: Objects that contain translations in the disabled language must be modified and saved. You can use the Search dialog box from the **Tools**

menu in Developer to locate objects that have translations in a given language. In the dialog box, on the International tab, click **Help** for details on setting up a search for these objects.

- For the entire metadata: Duplicate the project after the language has been removed, and do not include the translated strings in the duplicated project.

3 For objects that had the disabled language as their default language, the following scenarios occur. The scenarios assume the project defaults to English, and the French language is disabled for the project:

- If the object's default language is French, and the object contains both English and French translations, then, after French is disabled from the project, the object will only display the English translation. The object's default language automatically changes to English.
- If the object's default language is French and the object contains only French translations, then, after French is disabled from the project, the French translation will be displayed but will be treated by the system as if it were English. The object's default language automatically changes to English.

  For both scenarios above: If you later re-enable French for the project, the object's default language automatically changes back to French as long as no changes were made and saved for the object while the object had English as its default language. If changes were made and saved to the object while it had English as its default language, and you want to return the object's default language back to French, you can do so manually: right-click the object, select **Properties**, select **Internationalization** on the left, and choose a new default language.

Applying security and specialized translator user roles for languages

Each language in MicroStrategy is represented by a specific MicroStrategy object. You can apply security to a language in MicroStrategy by using the language object's ACLs, which permit or deny specific use of an object.

You also use the language object's ACLs in combination with MicroStrategy user privileges to create a translator or linguist role. This type of role allows a user to translate terms for an object in a given language, but keeps that user from making changes to the object's translations in other languages or making changes to the object's name and description in the object's default language.

Maintaining language objects and controlling security

Each language that is part of your MicroStrategy system (whether out of the box or languages you have added) exists as an object that can be edited, can have ACLs (security) set on it, can have its name translated, and so on.

ACLs can be used on a language object to control user access to certain languages. You can take advantage of this feature to allow users to serve themselves in terms of choosing

language preferences, while restricting them from languages that may not be supported for areas of the software the user commonly uses.

For example, you can create 2 groups of users and provide Group 1 with browse and use access to the English language object and the French language object, and provide Group 2 with browse and use access to Spanish only. In this scenario, users in Group 2 can only choose Spanish as their language preference, and can only access Spanish data from your warehouse. If an object which is otherwise available to Group 2 users does not have a Spanish translation, Group 2 users will be able to access that object in the project's default language (which may be English, French, or any other language.)

To access a language object

1. In Developer, from the Folder List on the left, within the appropriate project source, expand **Administration**, then expand **Configuration Managers**.
2. Select **Languages**. The language objects are displayed on the right side of Developer.
3. Right-click any language object to edit or otherwise maintain that object. Click **Help** for details on each interface you access.

Creating translator roles

You can set up MicroStrategy so that certain users can translate object names and descriptions into a given language. At the same time, you can restrict these users from changing the object's translations in languages other than the one they are translating, and from making any other changes to the object.

When an object is translated or an existing translation is edited, the object's version ID and modification timestamp are changed. This allows you to easily identify the latest translated objects when merging objects across projects or environments.

Creating a translator or linguist role can be useful if you have a translator who needs access to an object to translate it, but who should not have the ability to make any other changes to the object.

A common approach to setting up the MicroStrategy environment to support this type of user role requires creating a MicroStrategy user account specifically for each translator, allowing certain privileges to that user, and setting ACLs on one or more language objects to allow access to a given language for translation purposes. The steps below provide a common approach to setting up your system to support translator roles. The end goal is to create a list of user accounts made up of translators who have a limited set of permissions in MicroStrategy to translate a project's objects (schema objects, application objects, report/document objects), without the ability to write to any object or make changes to an object.

You can modify this approach to customize your language object security as it fits your specific needs. Suggestions are provided after the steps, to modify the translator role setup for specific situations.

The following terms are used:

- Source language: The object's default language
- Reference language: Any language other than the source language which the translator needs to translate from
- Target language: Any language other than the source language which the translator needs to translate to

To create a translator role

Create a user account for each translator

1 Create a user account for each translator.

   - Grant each user the Use Developer privilege, in the Analyst privilege group.
   - Grant each user the Use Translation Editor privilege, in the Common privilege group.
   - Grant each user the Use Translation Editor Bypass privilege, in the Developer privilege group.

     This privilege allows the user to use the Translation Editor to change an object's name and/or description for a given language, and does not require the user to have Write access to the object whose name/description is being changed (the system bypasses the normal check for Write access).

   For steps on creating a user account and assigning privileges to a user account, see the *Setting Up User Security* chapter in the *System Administrator Guide*.

Allow each translator user to add/edit translations for a given language

2 Grant the View permission on the ACL (access control list) for a language object to the user account that is allowed to translate objects into that language. This permission should be granted to the target language. The View permission allows a user to:

   - See an object's name and description in the source language (the object's default language) as well as in all other languages.
   - Translate object names and descriptions in the language the user has View permission for.

   To grant the View permission for a language object, use the following substeps:

   a In the Folder List, within the appropriate project source, expand **Administration**, then **Configuration Managers**, and select **Languages**.

   b Right-click a language from the list of language objects, and select **Properties**.

   c On the left, select **Security**.

d On the right, click **Add** to add the appropriate user account to the security for this language. Navigate to the appropriate translator user, select the user, and click **OK**.

e Click the field in the Permissions column next to the newly added user and select **View**.

A user who has View permissions to a language will be able to add or modify translations in that language using the Translation Editor. Translating an object's name/description in the source language (the object's default language) is equivalent to renaming the object. This may not be desirable, especially for schema objects. To prevent this, be sure that the View permission is not granted to the source language (the default language) of the objects that will be translated.

Allow translators to view translations in specific reference languages

3 Grant read-only access to one or more reference languages by granting the Browse and Read permissions (ACL permissions on the language object) for those languages that the translator needs to be able to view. Granting Browse and Read permission to a user for a language allows the translator to be able to see object names and descriptions in that language, but not to translate or otherwise make changes to object names/descriptions in that language. Read-only permission is generally granted to the source language (the object's default language), so that the source language can be used as the reference language during translation.

Allowing translators to see translations for an object in a language other than just the source language can provide translators useful context during translation, and is necessary if a translator needs to see a reference language that is different from the source language.

Use the following substeps:

a In the Folder List, within the appropriate project source, expand **Administration**, then **Configuration Managers**, and select **Languages**.

b Right-click a language from the list of language objects, and select **Properties**.

c On the left, select **Security**.

d On the right, click **Add** to add the appropriate user account to the security for this language. Navigate to the appropriate translator user, select the user, and click **Custom**.

e Click the field in the Permissions column next to the newly added user and select **Browse** and **Read**.

Be sure you do not grant the Use permission on any language object that represents a language you do not want the translator to be able to make changes to.

f Repeat these substeps for any other languages in the list of language objects that you want this user to be able to see.

To deny a translator the ability to see an object's name and description in a given language, assign the user the Deny All privilege for the language object(s) that the user should not be able to see or add/edit translations for.

Minimum requirements and additional options for creating a translator role

- The following table shows the minimum privileges and permissions that a user needs to be able to view a language and to translate schema objects, application objects, and report/document objects in a MicroStrategy project:

To View an Object's Name and Description in a Given Language	To Translate an Object's Name and Description in a Given Language
• The Use Developer privilege, in the Analyst privilege group.	• The Use Developer privilege, in the Analyst privilege group.
• The Use Translation Editor privilege, in the Common privilege group.	• The Use Translation Editor privilege, in the Common privilege group.
	• The Use Translation Editor Bypass privilege, in the Developer privilege group.
• The Browse permission on the language object that the translator will be translating into, and on a reference language object.	• The Browse permission on the language object that the translator will be translating into, and on a reference language object.
• The Read permission on the language object that the translator will be translating into, and on a reference language object.	• The Read permission on the language object that the translator will be translating into, and on a reference language object.
	• The Use permission on the language object that the translator will be translating into.

Be sure you do not grant the Use permission on any language object that represents a language you do not want the translator to be able to make changes to.

- To provide a translator the greatest possible context for objects:
  - Allow the translator user to see an object's name and definition in the source language and in any other language that the object uses, as well as the translator's target language. To do this, grant the translator user the Browse and Read permissions for each language object listed in **Administration** > **Configuration Managers** > **Languages**. The Browse and Read permissions allow the user to see translations in the Translation Editor but not edit the translation strings.
  - Grant the user privileges to access the object within the various Developer object editors. These privileges allow the user to execute the object so that it opens within its appropriate editor, thus displaying additional detail about the object. Access can allow context such as seeing a string as it appears within a dashboard; a metric's expression/formula; an attribute's forms and the data warehouse

tables that the data comes from; and so on. For example, in the User Editor, grant the translator the Execute Document and Use Report Editor privileges from the Analyst privilege group. Also grant Use Custom Group Editor, Use Metric Editor, Use Filter Editor, and so on, from the Developer privilege group.

- To deny a translator the ability to see an object's name and description in any language except the source language and the language that the translator has permission to Browse, Read, and Use, grant the user the Deny All privilege for the language objects that the user should not be able to see.

  For example, if you grant a translator Browse, Read, and Use permissions for the French language object, Browse and Read permissions for the object's default language, and Deny All for all other languages, the translator will only see the French translations column and the default language column in the Translations Editor in Developer.

  However, be aware that this limits the translator to only being able to use the object's default language as his reference language. If the translator can benefit from seeing context in other languages, it is not recommended to Deny All for other languages.

- You can create a security role to support per-project translator access. A security role is a set of project-level privileges. You can then assign the security role to individual users or groups. A user can have different security roles in different projects. For example, a user may have a Translator security role for the project she is supposed to translate, but the normal User security role in all other projects. Security roles are assigned to users or groups on a project-by-project basis.

  Because security roles are project-level roles, setting up translation based on security roles does not allow for the translation of configuration objects, such as database instances, schedules, events, and any other object that exists at the project source level. A translator can be set up to translate configuration objects using the information in the next bullet.

- To allow a translator to translate configuration objects (such as user and group descriptions, database instance names and descriptions, schedule and event names and descriptions, and any other objects that can be accessed by all projects in a project source), grant the translator the Use Translation Editor Bypass privilege at the user level (rather than at the project level). Also, grant the translator user the following privileges in the User Editor, which allow the user to access the various configuration object managers in the Administration folder in Developer:

  - Create and edit database instances
  - Create and edit database logins
  - Create and edit schedules and events
  - Create and edit security filters
  - Create and edit security roles
  - Create and edit users and groups
  - Create configuration objects

- To allow users to translate objects using MicroStrategy's bulk translation tool, the Repository Translation Wizard, grant the user the Use Repository Translation Wizard privilege.

  If this privilege is assigned, be aware that the user will be able to export strings and import translations for those strings in all languages that the project supports. This is true no matter what other language restrictions are applied.

3

MULTI-TENANT ENVIRONMENTS: OBJECT NAME PERSONALIZATION

In a multi-tenant setup, different organizations share a single MicroStrategy environment to accomplish their reporting needs. This chapter shows you how to use MicroStrategy to personalize object names in a project in your MicroStrategy environment, to support a multi-tenant setup.

Attribute and metric names in a project's metadata are made relevant to each tenant using object name personalization. Every object can have a different name stored to support each tenant who uses that object in their reporting. Each tenant's users see only those object names assigned to their organization. If there is no specific tenant name assigned to an object that is viewable by the tenant organization, its users see the base object name.

For example, you have an attribute stored in the metadata repository, with a base name of Inventory Date. This metadata object will appear on reports accessed by users in Organization A and Organization B. You can use object name personalization to configure MicroStrategy to automatically display the object to Organization A with the name Date In Inventory, and display the same object to Organization B with the name Date First Purchased.

Object name personalization involves exporting object strings to a location where they can be updated with tenant-specific names, and importing the new object strings back into the metadata repository. You can also provide new names for individual objects one at a time, using the Object Translation Editor in Developer.

For steps to perform these procedures, see *Renaming metadata objects, page 282*.

How a tenant language differs from a standard language

A tenant language is a set of objects that use the names appropriate for a given tenant. A tenant language appears in MicroStrategy exactly like any other language. The tenant

language's ID is the only property that differentiates a tenant language from a standard language; the system calculates the tenant language's ID based on the standard language's ID. For example, the language ID for standard English is 000 0409, while a tenant language based on standard English might be 0001 0409. Basing a tenant language on a standard language allows the system to provide the best match for all facets of the renamed interface, if one or more parts of the interface are not renamed for the tenant language.

You can create up to 255 tenant languages based on a standard language. For example, using English-US as the base language, you can create 255 tenant languages based on English-US. You can create another 255 tenant languages based on English-UK, and so on.

Granting user access to rename objects and view tenant languages

The procedures in this section will help you modify existing MicroStrategy projects to support metadata object renaming and tenant languages.

Allowing access to objects for renaming

To perform object renaming, access to specific objects is controlled primarily through access control lists (ACLs). You can allow permissions to specific users for each object that needs to be renamed, or for each tenant language (a set of objects for a given tenant).

Access to add or modify an object name

By default, administrators and object owners can rename an object or modify an existing object name. Use ACLs to provide other users Write access to an object, if other users need to rename that object. To change ACL permissions, right-click the object and select **Properties**, then select **Security** on the left. For details on each ACL and what access it allows, click **Help**.

You can also provide a user with the Use Repository Translation Wizard privilege, within the Object Manager set of privileges. This allows a user to perform the necessary steps to rename strings in bulk, for all tenants, without giving the user the ability to modify an object in any other way. To change a privilege, open the user in the User Editor and select **Project Access** on the left.

Access to select or enable a tenant's object names

By default, MicroStrategy users are provided with appropriate privileges to Browse and Use a tenant's objects, such that analysts can select a tenant language (the set of objects that use the tenant's names) as their display preference if that tenant language has been enabled for a project. Project administrators can enable any tenant language available in the system.

You can modify these default privileges for a specific user role or a specific tenant language.

To modify access to a tenant's set of object names (tenant language)

1. In the Folder List on the left, within the appropriate project source, expand **Administration**.
2. Expand **Configuration Managers**, then select **Languages**.
3. All tenant languages are listed on the right. To change ACL permissions for a tenant language, right-click the object and select **Properties**.
4. Select **Security** on the left. For details on each ACL and what access it allows, click **Help**.

Renaming metadata objects

Objects that can be renamed are stored in the MicroStrategy metadata. These objects include metric names, report names, the Public Objects system folder, security role names, user group names, and so on. Software strings stored in the metadata include embedded text strings (embedded in an object's definition), such as prompt instructions, aliased names (which can be used in attributes, metrics, and custom groups), consolidation element names, custom group element names, graph titles, and threshold text.

Metadata objects do not include configuration objects (such as the user object), function names, data mart table names, and so on.

Begin object renaming using the following high-level steps:

1. Add tenant languages to the system, for each of your tenants. For steps, see *Adding a new tenant language to the system, page 282*.
2. Enable tenant languages for your project's metadata objects. For steps, see *Enabling and disabling tenant languages, page 283*.
3. Provide tenant-specific names for objects using the steps in *Renaming objects in your project, page 285*

Adding a new tenant language to the system

You can add new tenant languages to MicroStrategy. Once they are added, new tenant languages are then available to be enabled for a project.

Prerequisites

- You must have the Browse permission for the language object's ACL (access control list). For details on ACLs, see the *System Administration Guide*.

To add a new tenant language to the system

1. Log in to a project as a user with administrative privileges.
2. Right-click the project and select **Project Configuration.** The Project Configuration Editor opens.
3. On the left side of the Project Configuration Editor, expand **Language**, then select **Metadata**.
4. Click **Add.** The Available Languages dialog box opens.
5. Click **New**. The Languages Editor opens. Click **Help** for details on each option.
6. Click **OK**. The Project Configuration Editor closes.
7. Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Tenant languages can also be added using the Languages Configuration Manager, by expanding **Administration** within the appropriate project source in the Folder List, expanding **Configuration Managers**, and selecting **Language**.

After adding a new tenant language, enable the tenant language for the project. For steps, see *Enabling and disabling tenant languages, page 283*.

Enabling and disabling tenant languages

To support the display of a tenant's object names and descriptions, you must enable tenant languages for your project. The tenant languages you enable are those tenant languages you want to support for that project.

You can also disable tenant languages for a project.

Enabling tenant languages for a project

Prerequisites

- Gather a list of tenant languages used by filters and prompts in the project. These tenant languages should be enabled for the project, otherwise a report containing a filter or prompt in a tenant language not enabled for the project will not be able to execute successfully.

To enable tenant languages for a project

1. Log into the project as a user with Administrative privileges.
2. Right-click the project and select **Project Configuration**. The Project Configuration Editor opens.

3. On the left, expand **Language** and select **Metadata**.
4. Click **Add** to see a list of available tenant languages. The Available Languages dialog box opens. The list includes languages that have been added to the system.
5. Select the check boxes for the tenant languages that you want to enable for this project.
   - Enabled tenant languages will appear in the Repository Translation Wizard for string and object renaming, as well as in Developer's My Preferences and Web's Preferences, for users to select their own preferred tenant language for the project.
   - Reports that contain filters or prompts in a tenant language will execute successfully if the project has that tenant language enabled.
6. Click **OK.** The Available Languages Dialog Box closes.
7. Select one of the tenant languages on the right side to be the default tenant language for this project. The default tenant language is used by the system to maintain object name uniqueness.

   This may have been set when the project was first created. If so, it will not be available to be selected here.

   Once the project default tenant language is set, it cannot be changed unless you duplicate the project and change the default tenant language of the duplicated project. Individual objects within a project can have their default tenant language changed.
8. Click **OK**.
9. Disconnect and reconnect to the project source.
10. Update the out-of-the-box MicroStrategy metadata objects. To do this, in Developer, right-click the project and select **Project Configuration**. Expand **Project Definition**, expand **Update**, select **Translations**, and click **Update**.

Disabling tenant languages for a project

You can use the steps below to disable a tenant language for a project. When a tenant language has been disabled from a project, that tenant language is no longer available for users to select as a tenant language preference, and the tenant language cannot be seen in any related interfaces, such as an object's Translation dialog box.

Any object names for the disabled tenant language are not removed from the metadata with these steps. Retaining the object names in the metadata allows you to enable the tenant language again later, and the object names will still exist. To remove object names in the disabled tenant language from the metadata, objects must be modified individually and saved.

To disable tenant languages in a project

1. Log in to a project as a user with administrative privileges.
2. Right-click the project and select **Project Configuration**. The Project Configuration Editor opens.
3. On the left side of the Project Configuration Editor, expand **Language**, then select **Metadata**.
4. On the right side, under **Selected Languages**, clear the check box for the tenant language that you want to disable for the project, and click **OK.** The Project Configuration Editor closes.

Renaming objects in your project

Renaming objects in a project involves providing new strings for metadata object names and descriptions.

There are two methods to rename metadata objects, depending on whether you want to rename a large number of objects or just one or two objects:

- **Rename a large number of objects**: Extract strings in bulk to a separate database, rename them, and import them back into MicroStrategy. The MicroStrategy Repository Translation Wizard is the recommended method to rename your metadata objects. Steps to access this tool are below.
- **Rename one or more objects in a folder**: Right-click the object and select **Translate**. Type the new name(s) for each tenant language this object supports, and click **OK**. To rename several objects, select them all while holding Shift or Ctrl, then right-click and select **Translate**. For details to use the Object Translation dialog box, click **Help**.

The rest of this section describes the method to rename object strings in bulk, using a separate database, with the Repository Translation Wizard.

The Repository Translation Wizard does not support renaming of configuration objects (such as the user object). It does support object descriptors, including embedded text. These are detailed in the introduction to *Renaming metadata objects, page 282*.

Object renaming involves the following high-level steps:

All of the procedures in this section assume that your projects have been prepared for object renaming. Preparation steps are in *Granting user access to rename objects and view tenant languages, page 281*.

1. Add and enable tenant languages for the metadata repository (see *Adding a new tenant language to the system, page 282* and *Enabling and disabling tenant languages, page 283*)
2. Export object strings to a location where they can be renamed (see *Extracting metadata object strings for renaming, page 286*)

3 Perform the renaming (see *Renaming objects in your project, page 285*)

4 Import the newly renamed object strings back into the metadata repository (see *Importing renamed strings from the database to the metadata, page 288*)

Prerequisites

- To allow users to rename objects using MicroStrategy's bulk translation tool, the Repository Translation Wizard, grant the user the Use Repository Translation Wizard privilege. If this privilege is assigned, be aware that the user will be able to export strings and import new names for those strings in all languages that the project supports. This is true no matter what other language restrictions are applied.

Extracting metadata object strings for renaming

The MicroStrategy Repository Translation Wizard supports Microsoft Access and Microsoft SQL Server databases as repositories where strings can be stored for renaming. The repository is where strings are extracted to and where the actual renaming process is performed.

You cannot extract strings from the project's default metadata language.

It is recommended that objects are not modified between the extraction process and the import process. This is especially important for objects with location-specific strings: attribute aliases, metric aliases, custom group elements, and document text boxes.

To extract a large number of object strings for renaming

1 Open the Repository Translation Wizard. To do this, from the **Start** menu, point to **All Programs**, then **MicroStrategy Tools**, then select **Repository Translation Wizard**. The Repository Translation Wizard opens.

2 Click **Next** to begin.

3 Click **Help** for details on each option in each page of the wizard.

   - To extract strings from the metadata, select the **Export Translations** option from the Metadata Repository page in the wizard.

Renaming metadata object strings in the database

The extraction process performed by the Repository Translation Wizard creates a table in the database, with the following columns:

- PROJECTID: This is the ID of the project from which the string is extracted.
- OBJECTID: This is the ID of the object from which the string is extracted.

- OBJECTTYPE: Each object is associated with a numeric code. For example, documents are represented by OBJECTTYPE code 55.
- EMBEDDEDID: An embedded object is an object contained inside another object, for example, a metric object that is part of a report object. If the string is extracted from an embedded object, the ID of this embedded object is stored in this column. The value 0 indicates that the string is not extracted from an embedded object.
- EMBEDDEDTYPE: This is a numeric representation of the type of the embedded object. The value 0 indicates that the string is not extracted from an embedded object.
- UNIQUEKEY: This is a key assigned to the extracted string to identify the string within the object.
- READABLEKEY: This is a description of the extracted string within the object, for example, Prompt Title, Prompt Description, Object Name, Template Subtotal Name, and so on. The READABLEKEY is a readable form of the UNIQUEKEY.
- LOCALEID: This indicates the tenant language of the extracted string in the TRANSLATION column.

  MicroStrategy uses locale IDs to uniquely identify tenant languages. MicroStrategy assigns a unique tenant language ID based on the base language that the tenant language is derived from.
- TRANSLATION: This is the column where the extracted string is stored.
- TRANSVERSIONID: This is the version ID of the object at the time of export.
- REFTRANSLATION: This column is used by translators. This column contains the extracted string in the translation reference language, which is selected by the user from the Repository Translation Wizard during export.

  This string is used only as a reference during the translation process. For example, if the translator is comfortable with the German language, you can set German as the translation reference language. The REFTRANSLATION column will then contain all the extracted strings in the German language.

  If no reference language string is available, the string from the object's primary language is exported so that this column is not empty for any string.
- STATUS: You can use this column to enter flags in the table to control which strings are imported back into the metadata. A flag is a character you type, for example, a letter, a number, or a special character (as long as it is allowed by your database). When you use the wizard to import the strings back into the metadata, you can identify this character for the system to use during the import process, to determine which strings to import.

  For example, if only some objects have been renamed, you may want to import only the completed ones. Or you may wish to import only those strings that were reviewed. You can flag the strings that were completed and are ready to be imported.
- OBJVERSIONID: This is the version ID of objects at the time of import.

- SYNCHFLAG: This is a system flag and is automatically generated during import. The following values are used:
  - 0: This means that the object has not been modified between extraction and import.
  - 1: This means that the object has been modified between extraction and import.
  - 2: This means that the object that is being imported is no longer present in the metadata.

  System flags are automatically applied to strings during the import process, so that you can view any string-specific information in the log file.
- LASTMODIFIED: This is the date and time when the strings were extracted.

Once the extraction process is complete, the strings in the database need to be renamed in the extraction table described above.

- If an object name is empty in a user's chosen project language preference, the system defaults to displaying the object's default name, so it is not necessary to rename objects that are not intended to be renamed. For details on language preferences, see *Selecting preferred languages for interfaces, reports, and objects, page 289*.
- If you performed a Search for Objects in the Repository Translation Tool, you may notice that the number of rows in the extraction table might not match the number of rows returned in the search results. This is because a search returns all objects that meet the search requirements; the search does not filter for only those items that can be renamed. Thus, for example, the search may return a row for the lookup table LU_YEAR, but the extraction process does not extract the LU_YEAR string because there is no reason to rename a lookup table's name. To determine whether an object's name can be renamed, right-click the object, select **Properties**, and look for the **International** option on the left. If this option is missing, the object is not supported for renaming.

To confirm that your new object names have successfully been imported back into the metadata, navigate to one of the renamed objects in Developer, right-click, and select **Properties**. On the left, select **International**, then click **Translate**. The table shows all names currently in the metadata for this object.

Importing renamed strings from the database to the metadata

After strings have been renamed in the database, they must be re-imported into the MicroStrategy metadata.

To import renamed strings

1. Open the Repository Translation Wizard. To do this, from the **Start** menu, point to **All Programs**, then **MicroStrategy Tools**, then select **Repository**

Translation Wizard. The Repository Translation Wizard opens.

2. Click **Next** to begin.
3. Click **Help** for details on each option in each page of the wizard.
    - To import strings from the database back into the metadata, select the **Import Translations** option from the Metadata Repository page in the wizard.

After the strings are imported back into the project, any objects that were modified while the renaming process was being performed, are automatically marked with a **1**. These object names should be checked for correctness.

Making tenant-specific data available to users

After you have performed the necessary steps to configure metadata object renaming, you can specify which tenant language(s) should be displayed for various users in the interface and in reports (both report objects and report results). You can specify language preferences at the project level and at the all-projects level. By selecting various levels of language preferences, you specify which language is preferred as a fallback if a first choice language is not available.

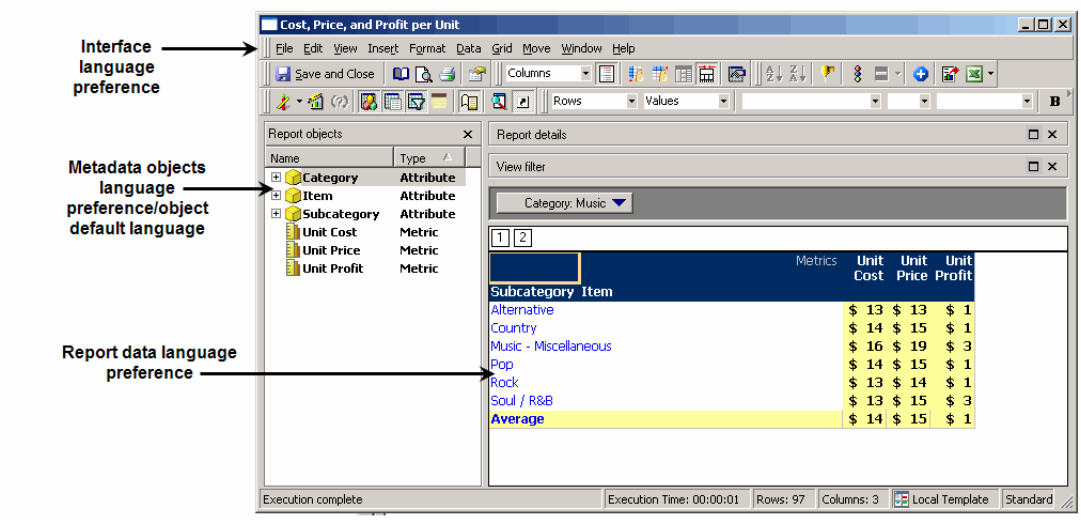
The following sections show you how to select language preferences based on various priority levels within the system, starting with a section that explains the priority levels:

- *Selecting preferred languages for interfaces, reports, and objects, page 289*
- *Selecting the Interface language preference, page 290*
- *Configuring metadata object and report data language preferences, page 292*
- *Selecting the object default language preference, page 302*

Selecting preferred languages for interfaces, reports, and objects

After renamed data is stored in your data warehouse and/or metadata database, and languages have been enabled for the project, you must specify which languages are the preferred languages for the project and the user. These selected languages are called language preferences.

The following image shows the different parts of the MicroStrategy environment that display renamed strings based on the language preferences:



The following language preferences can be configured:

- **Interface Language**: Determine the language that menu options, dialog box text, and so on, will display. For steps to set this preference, see *Selecting the Interface language preference, page 290*.
- **Metadata objects:** Determine the language that will be displayed for MicroStrategy objects that come from the metadata database, such as metric names, report names, system folder names, and so on. For steps to set this preference, see *Configuring metadata object and report data language preferences, page 292*.
- **Report data**: Determine the language that will be displayed for report results that come from your data warehouse, such as attribute element names. For steps to set this preference, see *Configuring metadata object and report data language preferences, page 292*.
- **Object default language**: Determine the fallback language for MicroStrategy objects. This language is used if a report is executed in a language that the object lacks a name for. For steps to set or change this default preference, see *Selecting the object default language preference, page 302*.

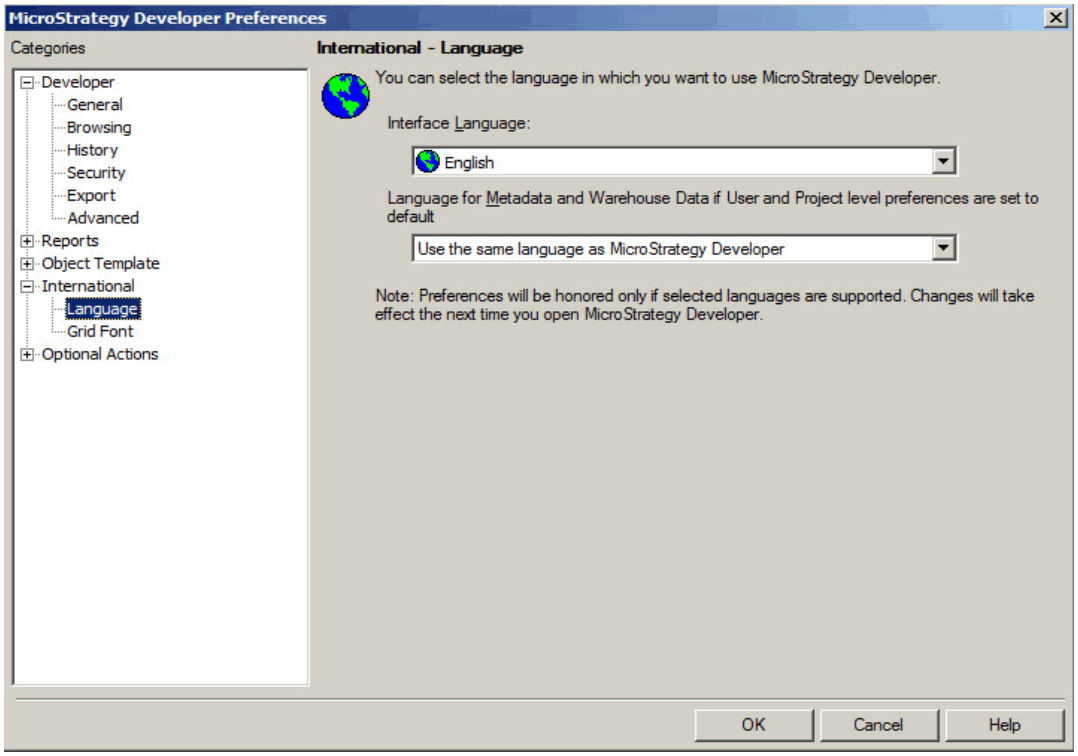
Each language preference can be configured independently of the others. However, for best performance it is recommended that you use a unified language display in Developer. For the purposes of multi-tenancy, this means that if the base language for a tenant language is English - US, all of the language selections for that tenant should be English - US with the exception of the Metadata Object language, which should be the tenant language.

Selecting the Interface language preference

The interface language preference determines what language Developer menus, editors, dialog boxes, monitors and managers, and other parts of the Developer software are displayed in. Use the steps below to set this preference.

Configuring the Interface language preference

1. In Developer, log in to the project.
2. From the **Tools** menu, select **Preferences**. The MicroStrategy Developer Preferences dialog box opens.
3. On the left, expand **International** and select **Language.** The International: Language dialog box is displayed, as shown below:



4. From the **Interface Language** drop-down list, select the language that you want to use as the interface default language

> The interface language preference can also be used to determine the language used for the metadata objects and report data, if the Developer level language preference is set to **Use the same language as MicroStrategy Developer.** For more information on the Developer level language preference, see *Selecting the Developer level language preference, page 298*.

5. Select **OK.** The Developer Preferences dialog box closes.
6. Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Configuring metadata object and report data language preferences

There are several levels at which metadata and report data languages can be specified in MicroStrategy. Lower level languages are used by the system automatically if a higher level language is unavailable. This ensures that end users see an appropriate language in all situations.

Language preferences can be set at six different levels, from highest priority to lowest. The language that is set at the highest level is the language that is always displayed, if it is available. If that language does not exist or is not available in the metadata or the data warehouse, the next highest level language preference is used.

If a language preference is not specified, or is set to Default, MicroStrategy automatically uses the next lower priority language preference. If none of these language preferences are set, the interface preferred language is used.

When an object is created, its default object language is automatically set to match the creator's metadata language preference. If the creator has his metadata language preference set to Default, the new object's default language is decided based on the rules in this section: the system will first try to use a default language for all users of the project, then a language preference set for all users of Developer, then the default language set for the project (as shown in the table below).

The following table describes each level, from highest priority to lowest priority, and points to information on how to set the language preference at each level.

- End user preference settings override any administrator preference settings, if the two settings conflict.
- Distribution Services deliveries are one exception to the hierarchy below. For details, see *Selecting the Machine level language preference, page 300*.

Language Preference Level (highest to lowest priority)	Description	Setting Location for End Users	Setting Location for Administrators
User-Project level	The language preference for a user for a specific project.	Web: From the MicroStrategy icon, select **Preferences**. Developer: From the **Tools** menu, select **My Preferences**.	Set in the User Language Preference Manager. See *Selecting the User-Project level language preference, page 294*.
User-All Projects level	The language preference for a user for all projects.	Web: From the MicroStrategy icon, select **Preferences**. Developer: From the **Tools** menu, select **My Preferences**.	Set in the User Editor. See *Selecting the User-All Projects level language preference, page 295*.
Project-All Users level	The language preference for all	Not applicable.	In the Project

Language Preference Level (highest to lowest priority)	Description	Setting Location for End Users	Setting Location for Administrators
	users in a specific project.		Configuration Editor, expand **Languages**, select **User Preferences**. See *Selecting the All Users In Project level language preference, page 296*.
Developer level	The interface language preference for all users of Developer on that machine, for all projects.	Set in the Developer Preferences dialog box. For steps to specify this language, see *Selecting the Developer level language preference, page 298*.	Set in the Developer Preferences dialog box. For steps to specify this language, see *Selecting the Developer level language preference, page 298*.
Machine level	The language preference for all users on a given machine.	On the user's machine and within the user's browser settings.	On the user's machine and within the user's browser settings. For steps to specify this language, see *Selecting the Machine level language preference, page 300*.
Project Default level	This is the project default language set for MDI. It is the language preference for all users connected to the metadata.	Not applicable.	Set in the Project Configuration Editor. For steps to specify this language, see *Configuring the Project Default level language preference, page 300*.

These language preferences apply to strings renamed in both the metadata and the data warehouse. However, MicroStrategy handles missing languages differently, depending upon whether the string is renamed in the metadata or the data warehouse:

- **Metadata**: When a name for an object in the metadata is missing in the preferred language, the object default language preference is used. For more information about the object default language preference, see *Selecting the object default language preference, page 302*.
- **Data warehouse**: When a name for data in the data warehouse is missing in the preferred language (the column or table is present in the data warehouse but is empty), the report returns no data.

The following sections provide steps to configure each preference level, starting from the highest priority and ending at the lowest priority.

Selecting the User-Project level language preference

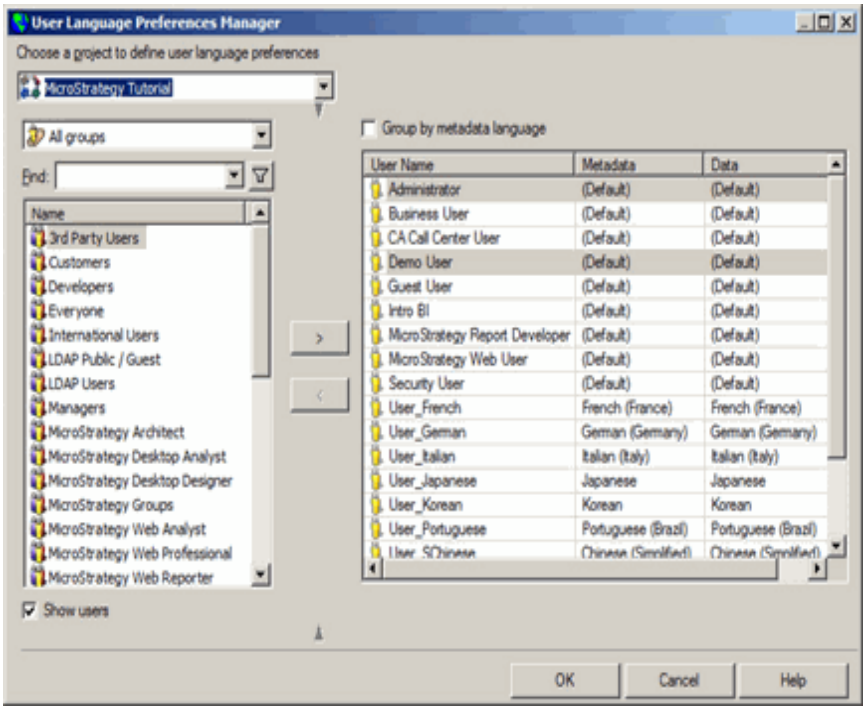
The User-Project Level language preference is the language preference for a given user for a specified project. It is the highest priority language setting; to see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 292*.

This preference is specified in the User Language Preference Manager in Developer. Use the steps below to set this preference.

If an object has an empty name in a user's chosen project language preference, the system defaults to displaying the object's default language, so it is not necessary to add names for objects that are not intended to be renamed.

Selecting the User-Project level language preference

1. Log in to Developer as a user with Administrative privileges.
2. Right-click the project that you want to set the language preference for and select **Project Configuration**. The Project Configuration Editor opens.
3. On the left side of the Project Configuration Editor**,** expand **Languages**, and select **User Preferences**.
4. On the right side, under **User Language Preference Manager**, click **Modify**. The User Language Preference Manager opens, shown below:



5. In the **Choose a project to define user language preferences** drop down menu at the top left, select the appropriate project.

6 Select the users from the list on the left side of the User Language Preferences Manager that you want to change the User-Project level language preference for, and click > to add them to the list on the right. You can narrow the list of users displayed on the left by doing one of the following:

- To search for users in a specific user group, select the group from the drop-down menu that is under the **Choose a project to define user language preferences** drop-down menu.
- To search for users containing a certain text string, type the text string in the **Find** field, and click the **Filter** icon: 

  This returns a list of users matching the text string you typed.

Previous strings you have typed into the **Find** field can be accessed again by expanding the **Find** drop-down menu.

7 On the right side, select the user(s) that you want to change the User-Project level preferred language for, and do the following:

You can select more than one user by holding CTRL.

- Select the desired language to be applied to renamed metadata objects from the drop-down menu in the **Metadata** column. This language will be displayed for the selected user(s) when connecting to the selected project.
- Select the desired language to be applied to report results from the drop-down menu in the **Data** column. This language will be displayed for the selected user (s) when connecting to the selected project.

8 Click **OK.** The preferences are saved and the User Language Preferences Manager closes.

Once the user language preferences have been saved, users can no longer be removed from the **Selected** list.

9 Click **OK.** The Project Configuration Editor closes.

10 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the User-All Projects level language preference

The User-All Projects level language preference determines what language will be applied to all projects that a specific user sees when connected to a project source, unless a higher priority language preference has been specified for the user. Use the steps below to set this preference.

If the User-Project language preference is specified for the user, the user will see the User-All Projects language only if the User-Project language is not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 292*.

Selecting the User-All Projects level language preference

1. Log in to Developer as a user with Administrative privileges.
2. In the Folder List on the left, within the appropriate project source, expand **Administration,** expand **User Manager**, and navigate to the user that you want to set the language preference for.
3. Double-click the user. The User Editor opens.
4. On the left side of the User Editor, expand the **International** category and select **Language.**
5. On the right side of the User Editor, do the following:
   - Select the language that you want to be applied to renamed metadata strings from the **Default metadata language preference for this user** drop-down menu.
   - Select the language that you want to be applied to renamed data warehouse strings from the **Default data language preference for this user** drop-down menu.
6. Click **OK.** The User Editor closes.
7. Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the All Users In Project level language preference

The All Users In Project level language preference determines the language that will be displayed for all users that connect to a project, unless a higher priority language is specified for the user. Use the steps below to set this preference.

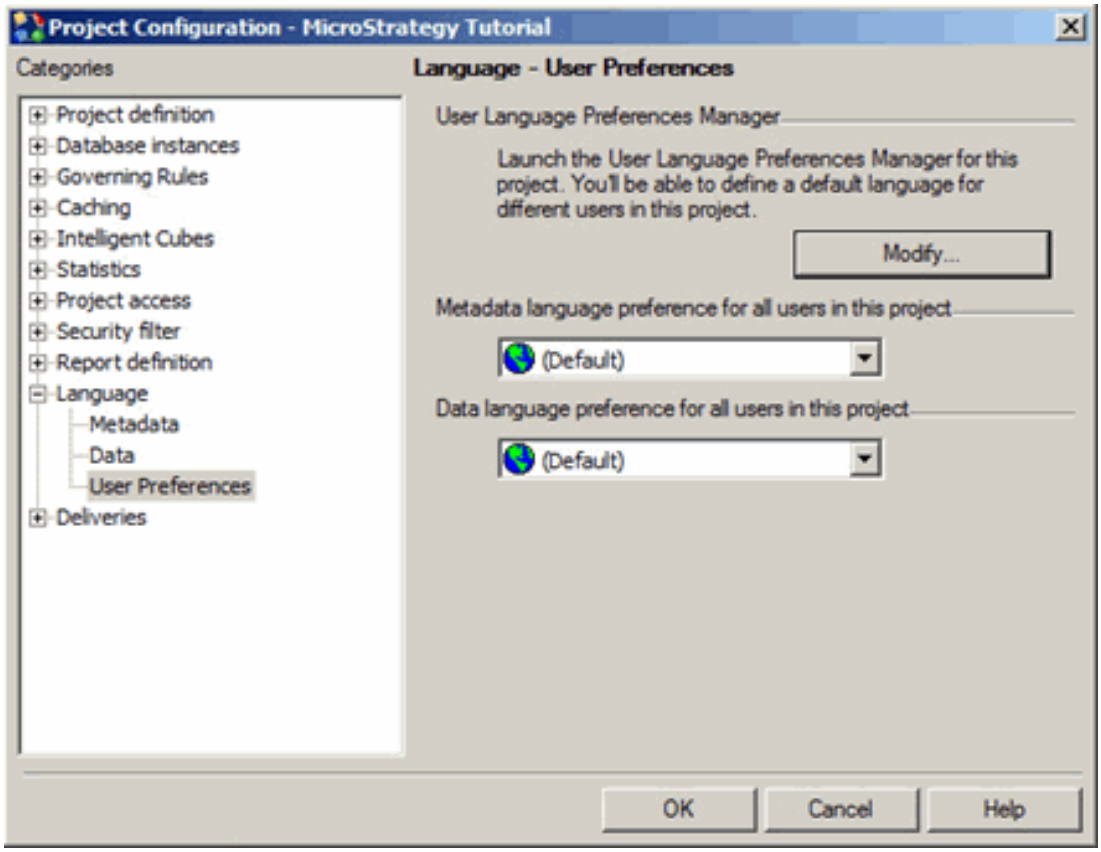
If the User-Project or User-All Projects language preferences are specified for the user, the user will see the All Users In Project language only if the other two language preferences are not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 292*.

Selecting the All Users In Project level language preference

1. Log in to Developer as a user with Administrative privileges.

2 In the Folder List on the left, select the project. From the **Administration** menu, select **Projects,** then **Project Configuration.** The Project Configuration Editor opens.

3 On the left side of the Project Configuration Editor, expand **Language** and select **User Preferences**. The Language-User Preferences dialog box is displayed, as shown below:



4 Do the following:

- From the **Metadata language preference for all users in this project** drop-down menu, select the language that you want to be displayed for metadata object names in this project.
- From the **Data language preference for all users in this project** drop-down menu, select the language that you want to be displayed for report results in this project.

5 Click **OK.** The Project Configuration Editor closes.

6 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the Developer level language preference

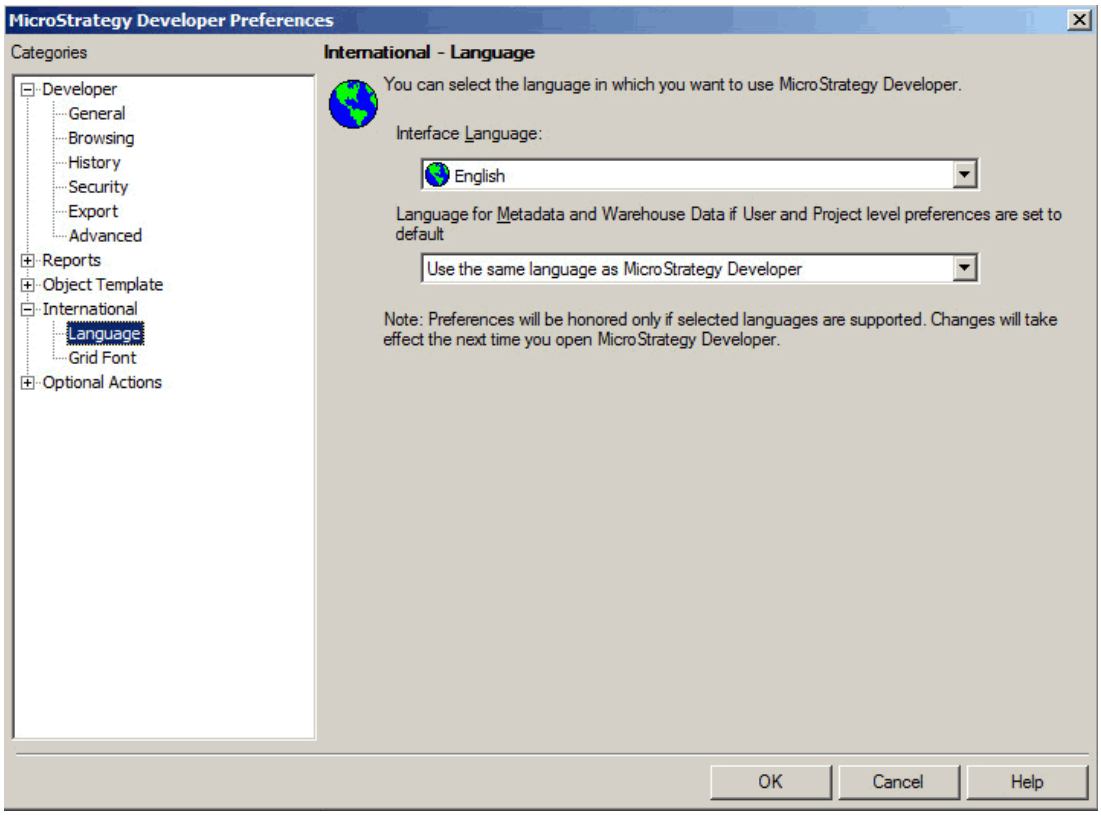
The Developer level language preference determines the default language for all objects displayed within Developer, unless a higher priority language preference has been specified. This is the same as the interface preference.

If the User-Project, User-All Projects, or All Users In Project language preferences are specified, the user will see the Developer language only if the other three language preferences are not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 292*.

This language preference must be configured to match one of two other language preferences: the Interface language preference or the Machine level language preference. For information about the Interface language preference, see *Selecting the Interface language preference, page 290*. For information about the Machine level language preference, see *Selecting the Machine level language preference, page 300*

Selecting the Developer level language preference

1. Log in to Developer as a user with Administrative privileges.
2. From the **Tools** menu, select **Preferences.** The MicroStrategy Developer Preferences dialog box opens.
3. Expand the **International** category and select **Language.** The International - Language dialog box opens, as shown below:



4 Select one of the following from the **Language for metadata and warehouse data if user and project level preferences are set to default** drop-down menu.

- If you want the Developer language preference to be the same as the Interface language preference, select **Use the same language as MicroStrategy Developer.** For information about configuring the Interface language preference, see ***Selecting the Interface language preference, page 290*****.**
- If you want the Developer language preference to be the same as the Machine-level language preference, select **Use language from Regional Settings**. For information about configuring the Machine-level language preference, see ***Selecting the Machine level language preference, page 300*****.**

5 Select the language that you want to use as the default Developer interface language from the **Interface Language** drop-down menu.

6 Click **OK.** The Developer Preferences dialog box closes.

7 Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the Machine level language preference

This preference determines the language that is used on all objects on the local machine. MicroStrategy Web uses the language that is specified in the user's web browser if a language is not specified at a level higher than this one.

- If the User-Project, User-All Projects, All Users In Project, or Developer language preferences are specified, the user will see the Machine language only if the other four language preferences are not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 292*.
- A MicroStrategy Distribution Services delivery (such as an email, file, or printer delivery) uses a different language resolution logic: If the User-Project, User-All Projects, All Users in Project, and Developer languages are not able to be displayed, the delivery defaults to the Project Default level language preference, followed by the Machine level language preference. This is because Distribution Services runs without a client session in the Intelligence Server machine; if the Machine level language took precedence, all users receiving delivered content would receive that content using the Intelligence Server machine's language. Instead, the project's default language is the fallback language for Distribution Services deliveries.

To select the Machine level language preference on a Windows machine, from the **Start** menu, select **Control Panel**, then **Regional and Language Options**. Consult your machine's Help for details on using the language options.

Configuring the Project Default level language preference

This language preference specifies the default language for the project. This language preference has the lowest priority in determining the language display. Use the steps below to set this preference.

- If the User-Project, User-All Projects, All Users In Project, Developer, or Machine-level language preferences are specified, the user will see the Project Default language only if the other five language preferences are not available. To see the hierarchy of language preference priorities, see the table in *Configuring metadata object and report data language preferences, page 292*.
- A MicroStrategy Distribution Services delivery (such as an email, file, or printer delivery) uses a different language resolution logic: If the User-Project, User-All Projects, All Users in Project, and Developer languages are not able to be displayed, the delivery defaults to the Project Default level language preference, followed by the Machine level language preference. This is because Distribution Services runs without a client session in the Intelligence Server machine; if the Machine level language took precedence, all users receiving delivered content would receive that content using the Intelligence Server machine's language. Instead, the project's default language is the fallback language for Distribution Services deliveries.

Selecting the Project Default language preference

The project default language is selected either when a project is first created, or the first time metadata languages are enabled for the project. It cannot be changed after that point. The following steps assume the project default language has not yet been selected.

1. Log in to the project as a user with Administrative privileges.
2. Select the project for which you want to set the default preferred language.
3. From the **Administration** menu, select **Projects**, then **Project Configuration.** The Project Configuration Editor opens.
4. On the left side of the Project Configuration Editor, expand **Language**. Do one or both of the following:
   - To specify the default metadata language for the project, select **Metadata** from the **Language** category. Then select **Default** for the desired language.
   - To specify the default data language for the project, select **Data** from the **Language** category. Then select **Default** for the desired language.
5. Click **OK.** The Project Configuration Editor closes.
6. Disconnect and reconnect to the project source so that your changes take effect. To do this, right-click the project source, select **Disconnect from Project Source**, then repeat this and select **Connect to Project Source**.

Selecting the object default language preference

Each MicroStrategy object can have its own default language. The object default language is used when the system cannot find or access a name for the object in the language specified as the user or project preference.

This preference is useful especially for personal objects, since most personal objects are only used in one language, the owner's language. The object default language can be set to any language supported by the project in which the object resides.

Some objects may not have their object default language preference set, for example, if objects are merged from an older MicroStrategy system that was not set up for multi-tenancy into an upgraded system that is set up for multi-tenancy. In this case, for those objects that do not have a default language, the system automatically assigns them the project's default language.

This is not true for newly created objects within an established multi-tenancy environment. Newly created objects are automatically assigned the creator's metadata language preference. For details on the metadata language, see *Configuring metadata object and report data language preferences, page 292*.

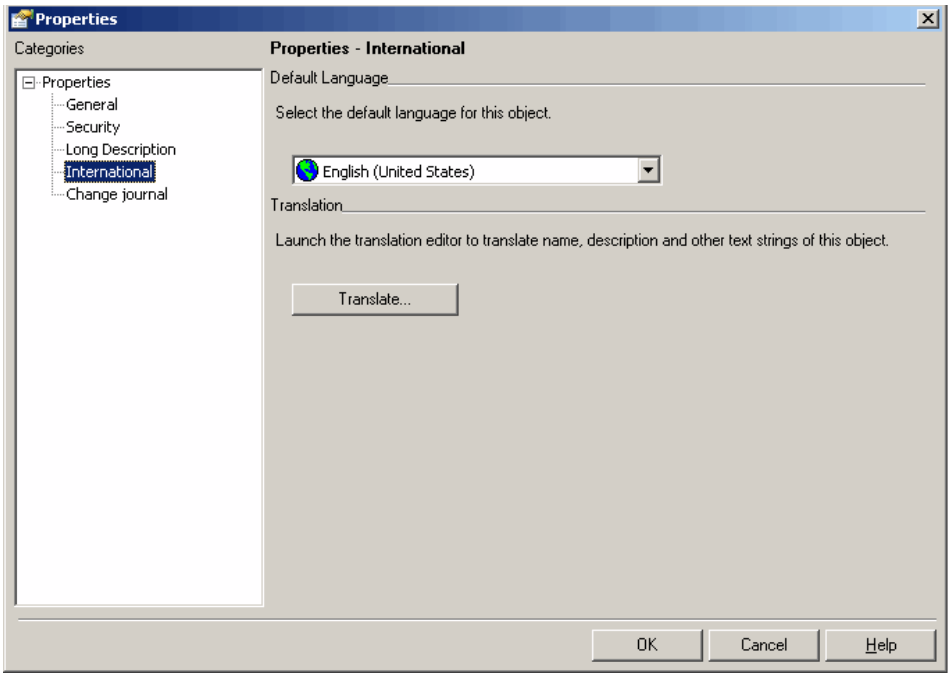
When duplicating a project, objects in the source that are set to take the project default language will take whatever the destination project's default language is.

Use the steps below to configure the object default language.

For the hierarchy of language preferences, see the table in *Configuring metadata object and report data language preferences, page 292*.

Configuring the object default language preference

1. Log in to the project source that contains the object as a user with administrative privileges.
2. Right-click the object and select **Properties.** The Properties dialog box opens.
   - You can set the default language for multiple objects by holding the **Ctrl** key while selecting multiple objects.
3. Select **International.** The Properties - International dialog box is displayed, as shown below:



If the International option is missing, the object is not supported for renaming. For example, there is no reason to rename a table name for a schema object (such as LU_YEAR), so this object does not have the International option available.

4 From the **Select the default language for the object** drop-down menu, select the default language for the object(s).

5 Click **OK.** The Properties dialog box closes.

Achieving the correct language display

The following table lists many of the locations where you might want to display a given language for users. It tells you where to configure the system so that the language is displayed or available for selection. For some language displays, there are different steps in Developer than in MicroStrategy Web.

Name or Language Display that You Want to Achieve	Where to Enable It
User changing own language	In Web: Click the **MicroStrategy** icon, then select **Preferences**. In Developer: From the **Tools** menu select **MicroStrategy Developer Preferences**. The list of languages to choose from comes from the languages enabled for a project; see *Enabling tenant languages for a project, page 283*.

Name or Language Display that You Want to Achieve	Where to Enable It
Default language preference for a particular user	In the User Editor, expand **International**, then select **Language**. An administrator needs the Use User Editor and Configure Language Settings privileges, and ACL permissions to modify the user object.
Default language for all users in a project	Right-click a project, select **Project Configuration**, expand **Language**, and select **User Preferences**.
Different default language for a single user in different projects	Right-click a project, select **Project Configuration**, expand **Language**, and select **User Preferences**.
Function names	Function names are not renamed. The MicroStrategy system expects function names to be retained.
An individual object	Use the Object Translation Editor. To access this, right-click the object and select **Translate**.
Subscriptions in a multi-tenant environment	Subscribed-to reports and documents behave like standard reports and documents, and are delivered in the tenant language selected in My Preferences or User Preferences.
Repository Translation Wizard list of available languages	Enable languages the project supports for metadata objects (see *Enabling tenant languages for a project, page 283*).
Metadata object names and descriptions (such as report names, metric names, system folder names, embedded descriptors such as attribute aliases, prompt, instructions, and so on)	See *Enabling tenant languages for a project, page 283*.
Configuration objects in Developer	Displayed according to User-Project level language preference. Set this by right-clicking the project, selecting **My Preferences**, selecting **International**, and setting the Metadata language for All Projects.
Project name and description	In the Project Configuration Editor, expand **Project Definition**, select **General**, click **Modify**, select **Internationalization**, then click **Translate**. You can type both a project name and a description in the Object Description field.
Enable a tenant language for a project to support	See *Adding a new tenant language to the system, page 305*. Then see *Enabling tenant languages for a project, page 283*. User adding the language must have Browse permission for that language object's ACL.
Searching the project	Searches are conducted in the user's preferred metadata language by default. A language-specific search can be conducted; open a project, then from the **Tools** menu select **Search for Objects**. On

Name or Language Display that You Want to Achieve	Where to Enable It
	the International tab, click **Help** for details on each option in the interface.
Project or object migration, or duplication	Object Manager, Project Merge, and the Project Duplication Wizard contain name-specific conflict resolution options for migrating renamed objects between projects. Click **Help** in any of these tools for details, or see the *Managing Your Projects* chapter in the *MicroStrategy System Administration Guide*.
Derived elements	In the Derived Element Editor, from the **File** menu, select **Properties**, then select **International**. For details to use the options, click **Help**.
MicroStrategy Office user interface and Excel format languages	In MicroStrategy Office, select **Options**, select **General**, then select **International**.

Maintaining your multi-tenant environment

You can add or remove tenant languages from your MicroStrategy system, and you can edit the object names in the system. This section also covers security and specialized user roles for object renaming.

Adding a new tenant language to the system

You can add new languages to MicroStrategy. Once they are added, new languages are then available to be enabled for a project. For steps to add a new tenant language, see *Adding a new tenant language to the system, page 282*.

Removing a tenant language from the system

A language cannot be removed from the system if it is being used by a project, that is, if it has been enabled to be supported for a project. To remove a tenant language from a project, that language must first be disabled from the project, as described in the steps below.

If a user has selected the language as a language preference, the preference will no longer be in effect once the language is disabled. The next lower priority language preference will take effect. To see the language preference priority hierarchy, see *Configuring metadata object and report data language preferences, page 292*.

To remove a tenant language from the system

1. Disable the tenant language from all projects in which it was enabled. To disable a metadata language from a project, see *Enabling and disabling tenant languages, page 283*.

2. For metadata languages, any names for the disabled language are not removed from the metadata with these steps. To remove names:

   - For individual objects: Objects that contain names for the disabled tenant language must be modified and saved. You can use the Search dialog box from the **Tools** menu in Developer to locate objects that have names for a given tenant. In the dialog box, on the International tab, click **Help** for details on setting up a search for these objects.
   - For the entire metadata: Duplicate the project after the tenant language has been removed, and do not include the renamed strings in the duplicated project.

3. For objects that had the disabled language as their default language, the following scenarios occur. The scenarios assume the project defaults to Tenant A's language, and Tenant B's language is disabled for the project:

   - If the object's default language is Tenant B's language, and the object has names for both Tenant A and Tenant B, then, after Tenant B's language is disabled from the project, the object will only display Tenant A's names. The object's default language automatically changes to Tenant A's language.
   - If the object's default language is Tenant B's language and the object contains only Tenant B's names, then, after Tenant B's language is disabled from the project, Tenant B's names will be displayed but will be treated by the system as if they belong to Tenant A's language. The object's default language automatically changes to Tenant A's language.

     For both scenarios above: If you later re-enable Tenant B's language for the project, the object's default language automatically changes back to Tenant B's language as long as no changes were made and saved for the object while the object had Tenant A's language as its default language. If changes were made and saved to the object while it used Tenant A's language as its default language, and you want to return the object's default language back to Tenant B's language, you can do so manually: right-click the object, select **Properties**, select **Internationalization** on the left, and choose a new default language.

4

LIST OF PRIVILEGES

This chapter provides reference information for privileges in MicroStrategy. For general information about using privileges and security roles, see the *Setting up User Security* chapter in the System Administration Guide.

This chapter includes the following:

- *Privileges for predefined security roles, page 308*
- *Privileges for out-of-the-box user groups, page 309*
- *List of privileges required for specific Flash Mode interactions, page 314*
- *List of all privileges, page 315*

  Client-based privileges:

  - *Web Reporter privileges, page 316*
  - *Web Analyst privileges, page 317*
  - *Web Professional privileges, page 320*
  - *MicroStrategy Desktop privileges, page 322*
  - *Common privileges, page 323*
  - *MicroStrategy Office privileges, page 324*
  - *MicroStrategy Mobile privileges, page 324*
  - *Deliveries privileges, page 325*
  - *Analyst privileges, page 326*
  - *Developer privileges, page 327*

  Administrator privileges:

  - *MultiSource Option privileges, page 326*
  - *Architect privileges, page 329*
  - *Object Manager privileges, page 329*
  - *Command Manager privileges, page 330*

- *Integrity Manager privileges, page 330*
- *Administration privileges, page 330*

Privileges for predefined security roles

The MicroStrategy product suite contains a number of predefined security roles for administrators. These roles makes it easy to delegate administrative tasks.

The predefined project administration roles apply project-level administrative privileges. The default privileges that are automatically granted for these out-of-the-box security roles are listed below.

Power Users have the following privileges:

Project Bulk Administrators	
• Schedule Request (in Common Privileges) • Assign security roles • Bypass all object security access checks	• Create and edit security filters • Edit project status • Web administration

Project Bulk Administrators have the following Object Manager privileges:

Project Bulk Administrators	
• Use Object Manager	• Use Repository Translation Wizard

Project Operations Administrators have the following privileges:

Project Operations Administrators	
• Schedule Request (in Common Privileges) • Administer Caches • Administer Cubes • Administer Jobs	• Administer Subscriptions • Administer User Connections • Idle And Resume Project • Load And Unload Project

Project Operations Monitors have the following privileges:

Project Operations Monitors	
• Administer Caches	• Monitor Cubes

Project Operations Monitors	
• Administer Jobs • Administer User Connections • Audit Change Journal • Idle And Resume Project • Load and Unload project • Monitor Caches	• Monitor History Lists • Monitor Jobs • Monitor Projects • Monitor Subscriptions • Monitor User Connections

Project Resource Settings Administrators have the following privileges:

Project Resource Settings Administrators	
• Configure Caches • Configure Governing • Configure Language Settings • Configure Project Basic • Configure Project Data Source	• Configure Statistics • Configure Subscription Settings • Edit Project Status • Web Administration

Project Security Administrators have the following privileges:

Project Security Administrators	
• Create Application Objects (in Common Privileges) • Assign Security Filters • Assign Security Roles	• Configure Connection Map • Configure Security Settings • Create And Edit Security Filters

Privileges for out-of-the-box user groups

The privileges that are automatically granted for out-of-the-box groups are listed below.

- All users and user groups are members of the **Everyone** group and inherit all privileges granted to that group.
- Installing the MicroStrategy Tutorial may change the default privileges granted for some of these groups.

The following MicroStrategy user groups have no default privileges other than those granted by the Everyone group:

- 3rd Party Users

- LDAP Public/Guest
- LDAP Users
- Public/Guest
- Warehouse Users

Privileges for the Everyone group

By default the **Everyone** group does not grant any privileges.

When a project is upgraded from MicroStrategy version 7.2.x or 7.5.x to version 9.0 or later, the Use Developer privilege in the Analyst privilege group is automatically granted to the Everyone group. This ensures that all users who were able to access Developer in previous versions can continue to do so.

Privileges for the MicroStrategy Architect group

The **MicroStrategy Architect** group grants all privileges in the Architect privilege group (see *Architect privileges, page 329*). In addition, it grants the following privileges:

- Create Schema Objects, in Common Privileges
- Import Table From Multiple Sources, in MultiSource Option

Privileges for the Developer groups

The default privileges that are automatically granted for the Developer groups are listed below.

The **Analyst** group grants all privileges in the Analyst privilege group (see *Analyst privileges, page 326*). In addition, it grants the following privileges from the Common Privileges privileges group:

Analyst	
• Drill Within Intelligent Cube • Add Notes • Create Application Object • Create Folder • Create Shortcut • Edit Notes	• Save Personal Answer • Schedule Request • Use Server Cache • Use Translation Editor • View Notes

The **Developer** group grants the following privileges:

- All privileges granted to the Analyst group.
- All privileges in the Developer privilege group (see *Developer privileges, page 327*), except for Create HTML Container and Use Bulk Export Editor.

- The privileges Create Application Objects and Create Schema Objects, in Common Privileges.

Privileges for the MicroStrategy Web groups

The default privileges that are automatically granted for the MicroStrategy Web groups are listed below.

The **MicroStrategy Web Reporter** group grants the following privileges:

- All privileges in the Web Reporter privilege group (see *Web Reporter privileges, page 316*).
- All privileges in the Common Privileges privilege group, except for Create Schema Objects and Edit Notes.

The **MicroStrategy Web Analyst** group grants the following privileges:

- All privileges granted to the MicroStrategy Web Reporter group.
- All privileges in the Web Analyst privilege group (see *Web Analyst privileges, page 317*).
- The following additional privileges:

MicroStrategy Web Analyst	
• Web Drill And Link (in Web Reporter) • Web Simultaneous Execution (in Web Reporter) • Web View History List (in Web Reporter)	• Create Application Objects (in Common Privileges) • Schedule Request (in Common Privileges) • Use Distribution Services (in Distribution Services)

Some of these privileges are also inherited from the groups that the Web Analyst group is a member of.

The **MicroStrategy Web Professional** group grants the following privileges:

- All privileges granted to the MicroStrategy Web Analyst group.
- All privileges in the Web Professional privilege group (see *Web Professional privileges, page 320*), except for Web Create HTML Container.
- The following additional privileges:

MicroStrategy Web Analyst	
• Web Drill And Link (in Web Reporter) • Web Simultaneous Execution (in Web Reporter) • Web View History List (in Web Reporter)	• Create Application Objects (in Common Privileges) • Schedule Request (in Common Privileges) • Use Distribution Services (in Distribution Services)

Some of these privileges are also inherited from the groups that the Web Professional group is a member of.

Privileges for the System Monitors groups

By default the **System Monitors** group does not grant any additional privileges. The default privileges that are automatically granted for the groups that are members of the System Monitors group are listed below. Unless otherwise specified, all privileges are from the Administration privilege group (see *Administration privileges, page 330*).

The **Narrowcast System Administrators** group does not grant any privileges by default.

The **Server Bulk Administrators** group grants the following privileges:

Server Bulk Administrators	
• Use Object Manager • Use Command Manager	• Use Repository Translation Wizard

The **Server Configuration Administrators** group grants the following privileges:

Server Configuration Administrators	
• Create And Edit Database Instances And Connections • Create And Edit Database Logins	• Create Configuration Objects • Create And Edit Transmitters And Devices (in Distribution Services)

The **Server Operations Administrators** group grants the following privileges:

Server Operations Administrators	
• Schedule Request (in Common Privileges) • Administer Caches • Administer Cluster • Administer Cubes • Administer Database Connections	• Administer Jobs • Administer Subscriptions • Administer User Connections • Fire Events • Idle And Resume Project • Load And Unload Project

The **Server Operations Monitors** group grants the following privileges:

Server Operations Monitors	
• Administer Caches • Administer Cluster • Administer Database Connections • Administer Jobs • Administer User Connections • Audit Change Journal • Idle And Resume Project • Load And Unload Project • Monitor Caches	• Monitor Cluster • Monitor Cubes • Monitor Database Connections • Monitor History Lists • Monitor Jobs • Monitor Projects • Monitor Subscriptions • Monitor User Connections

The **Server Resource Settings Administrators** group grants the following privileges:

Server Resource Settings Administrators	
• Configure Caches • Configure Governing • Configure Language Settings • Configure Project Basic • Configure Project Data Source • Configure Server Basic	• Configure Statistics • Configure Subscription Settings • Edit Project Status • Web Administration

The **Server Security Administrators** group grants the following privileges:

Server Security Administrators	
• Create Application Objects (in Common Privileges) • Assign Security Filters • Assign Security Roles • Configure Connection Map	• Configure Security Settings • Create And Edit Security Filters • Grant/Revoke Privilege

The **System Administrators** group grants all MicroStrategy privileges.

The **User Administrators** group grants the following privileges:

User Administrators	
• Configure Contacts Data Security Profile (in Distribution Services) • Assign Security Roles • Configure Group Membership • Create And Edit Contacts And Addresses • Create And Edit Security Roles	• Create And Edit Users And Groups • Create Configuration Objects • Enable User • Grant/Revoke Privilege • Link Users And Groups To External Accounts • Reset User Password

List of privileges required for specific Flash Mode interactions

Flash Mode is available only if your document was created before MicroStrategy 10.1 and was defined to display in Flash Mode. Beginning with 10.1, Presentation Mode replaces Flash Mode. For information on working with documents in Flash Mode and Presentation Mode, see the *MicroStrategy Web Help*.

Most document interactions in Flash Mode in MicroStrategy Web require the same privileges as interactions in other Web modes. That is, sorting requires the Web Sort privilege, using a link requires the Web Drill And Link privilege, and so on.

However, certain actions in Flash Mode require specific privileges. These actions and their corresponding privileges are listed below. For a detailed explanation of these actions, see the *MicroStrategy Web Help*.

Action	Privileges Required
Add To Grid	• Web Add/Remove Units To/From Grid In Document In Flash Mode
Drill	• Drill Within Intelligent Cube (in the Common privileges group) • Web Drill And Link • Web Drill On Metrics (to drill on metrics) • Web Advanced Drilling (to drill to low priority paths from a metric, attribute, or attribute element)
Move To Beginning	• Web Sort • Web Pivot Report
Filter	• Web Use View Filter Editor
Clear Filter	Either one of the following: • Web User View Filter Editor • Web Filter On Selections
Keep Only / Exclude	• Web Filter On Selections

List of all privileges

The sections below list all of the privileges available in the MicroStrategy environment. They are grouped by privilege type. To locate information about a specific privilege, use the Find In PDF feature of your PDF viewer to search for the name of that privilege.

To add or remove a privilege for a user or group, right-click that user or group and select **Edit** to open it in the User Editor or Group Editor. Select the **Project Access** category, and select or clear the check boxes for the privilege for any projects you want the privilege to apply to. For a more in-depth discussion of privileges, including instructions on using the **Project Access** category to assign privileges, see the *Setting Up User Security* chapter in the *MicroStrategy System Administration Guide*.

Client-based privileges are covered in these groups:

- *Web Reporter privileges, page 316*
- *Web Analyst privileges, page 317*
- *Web Professional privileges, page 320*
- *MicroStrategy Desktop privileges, page 322*
- *Common privileges, page 323*
- *MicroStrategy Office privileges, page 324*
- *MicroStrategy Mobile privileges, page 324*
- *Deliveries privileges, page 325*
- *Analyst privileges, page 326*
- *Developer privileges, page 327*

Administrator privileges are covered in these groups:

- *MultiSource Option privileges, page 326*
- *Architect privileges, page 329*
- *Object Manager privileges, page 329*
- *Command Manager privileges, page 330*
- *Integrity Manager privileges, page 330*
- *Administration privileges, page 330*

Privilege availability

Privileges are available to be assigned to users or groups depending on whether the appropriate license has been purchased for a product. A privilege is available if it is enabled in the User Editor, that is, if it can be selected and is not grayed out. If you have not purchased a license for a product, that product's privileges are grayed out in both the User Editor and the Security Role editor.

A privilege with the note “Server level only” can be granted only at the project source level. It cannot be granted for a specific project.

Certain privileges are marked with asterisks in the tables below, for the following reasons:

- Privileges marked with * are included only if you have OLAP Services installed as part of Intelligence Server.
- Privileges marked with ** are included only if you have Report Services installed.
- Privileges marked with *** are included only if you have Distribution Services installed.
- Privileges marked with **** are included only if you have Transaction Services installed.

These privileges are unavailable (greyed out) in the User Editor if you do not have a license for the appropriate MicroStrategy product. To determine your license information, use License Manager to check whether any of the specified products are available.

Web Reporter privileges

These privileges, along with the common privileges (see *Common privileges, page 323*), correspond to the basic functionality available in MicroStrategy Web. The predefined MicroStrategy Web Reporter group is assigned these privileges by default. License Manager counts any user who has any of these privileges, but none of the Web Analyst or Web Professional privileges, as a Web Reporter user.

MicroStrategy Web users that are licensed for MicroStrategy Report Services can view and interact with a document in Flash Mode, if the document was created before MicroStrategy 10.1 and was defined to display in Flash Mode. Certain interactions in Flash Mode have additional licensing requirements:

- Users are required to license MicroStrategy Web Analyst to pivot row or column position in a grid or cross-tabular grid of data in Flash Mode.
- Users are required to license MicroStrategy Web Professional to modify the properties of widgets used in a document in Flash Mode.

Beginning with 10.1, Presentation Mode replaces Flash Mode; newly created documents automatically display in Presentation Mode. For information on working with documents in Flash Mode and Presentation Mode, see the *MicroStrategy Web Help*.

Privilege	Allows the user to...
Web change view mode	Toggle between grid, graph, and grid & graph, to hide or show predefined totals, and to reset reports
Web configure toolbars	Use, show, and hide toolbars

Privilege	Allows the user to...
Web drill and link	Use links to view related data not shown in the original report results
Web export	Export report results
Web object search	Search for reports, documents, folders, filters, or templates
Web print mode	Display a printable version of a report or document
Web re-execute report against warehouse	Re-execute a report, hitting the warehouse rather than the server cache. If Intelligence Server caching is turned off and this is not granted, the re-execute button is removed.
Web simultaneous execution	Open multiple MicroStrategy Web sessions
Web sort	Sort report data by clicking on sort icons in column headings
Web subscribe to History list	Subscribe to periodic execution of reports and view their results via the History List A user with this privilege is also considered to have the Schedule Request privilege in Common Privileges.
Web switch page-by elements	Switch page-by elements for objects in the Page axis
Web use locked headers	Use the Lock Grid Headers feature
Web user	Connect to MicroStrategy using the Web interface If a user does not have this privilege for a project, that project is not shown in Web. If a user does not have this privilege for any projects available on the Web server, Web returns the error message, “No projects are available for this project source.”
Web: Change user preferences	Change some characteristics of page appearance and report results

Web Analyst privileges

These privileges correspond to the report analysis and creation functionality available to users of MicroStrategy Web. The predefined MicroStrategy Web Analyst group is assigned these privileges by default. The Web Analyst group also inherits all of the privileges assigned to the Web Reporter group. License Manager counts any user who has any of these privileges, but none of the Web Professional privileges, as a Web Analyst user.

MicroStrategy Web users that are licensed for MicroStrategy Report Services can view and interact with a document in Flash Mode, if the document was created before MicroStrategy 10.1 and was defined to display in Flash Mode. Certain interactions in Flash Mode have additional licensing requirements:

- Users are required to license MicroStrategy Web Analyst to pivot row or column position in a grid or cross-tabular grid of data in Flash Mode.

- Users are required to license MicroStrategy Web Professional to modify the properties of widgets used in a document in Flash Mode.

Beginning with 10.1, Presentation Mode replaces Flash Mode; newly created documents automatically display in Presentation Mode. For information on working with documents in Flash Mode and Presentation Mode, see the *MicroStrategy Web Help*.

Privilege	Allows the user to...
* Access Data (Files) from Local, URL, DropBox, Google Drive, Sample Files, Clipboard	Import data from files using the Connect to Your Data dialog
* Access data from Cloud App (Google Analytics, Salesforce Reports, Facebook, Twitter)	Import data from cloud data sources using the Connect to Your Data dialog
* Access Data from Databases, Google BigQuery, BigData, OLAP, BI tools	Import data from databases using the Connect to Your Data dialog
* Web add/remove units to/from grid in document in View Mode	Add objects to or remove objects from an existing grid report displayed in a document when in Presentation Mode
* Web create derived metrics	Create new calculations based on other metrics already on a base report
* Web define derived elements	Create derived elements in a report or document
* Web number formatting	Change number formatting on all metrics on grids A user with this privilege has access to the following formatting options: • From the Format menu, the Advanced Formatting option • The Advanced Formatting context menu (all options) • On the Format toolbar, the Area (metrics only), Type, and number style options Assigning this privilege also assigns the Use OLAP Services privilege (in the Common privileges group).
* Web publish Intelligent Cube	Publish an Intelligent Cube to Intelligence Server
* Web use Report Objects window	Use the Report Objects panel
* Web use View filter editor	Add or modify the view filter for a report
** Web create dashboard	Create a new dashboard
** Web dashboard design	Add visualizations and other objects (such as filters, text, and images) to a dashboard, and edit existing visualizations
** Web Edit dashboard	Edit a dashboard

Privilege	Allows the user to...
	If you do not have this permission, the Visualization Gallery and the Dashboard Datasets, Editor, Filter, and Properties panels are not displayed.
** Web save dashboard	Enables the Save button in Visual Insight Dashboards See Web save to My Reports or Web save to Shared Reports for the ability to save Reports and Documents.
*** Web create alert	Create a conditional subscription based on a threshold
*** Web subscribe to bulk export	Create a bulk export subscription
Import MicroStrategy file	To upload a dashboard to the server, for sharing the dashboard with other users
Set oAuth Parameters for Cloud App sources	Modify oAuth parameters for cloud sources from the Connect to Your Data dialog
Web add to History List	Add reports or documents to the History List (requires Web simultaneous execution privilege)
Web advanced drilling	Access advanced drill mode through the More Options link on the report results page
Web alias objects	Rename an object on a report
Web choose attribute form display	Use the Attribute Forms dialog box, see attribute forms in the Report Objects list, see the Attribute Forms context menu options, and pivot attribute forms
Web create new report	Access the Create Report folder and design reports, and run new reports from the folder where he or she has saved the report definition
Web drill on metrics	Drill on metrics
Web edit notes	Add and edit notes that have been added to a report or document
Web execute data mart report	Execute data mart reports
Web filter on selections	Filter on selected data. This privilege is also required for the Keep Only and Execute actions in Flash Mode.
Web manage objects	Delete folders, reports, and documents
Web modify Subtotals	Add, modify or delete subtotals on a Web report
Web pivot report	Move rows and columns up or down and left or right, to pivot from rows to columns and vice versa, and to move metrics, attributes, custom groups, and consolidations to the Page axis
Web report details	Access report and document information by clicking the Report details link on the report, History List, or Wait page

Privilege	Allows the user to...
Web report SQL	View the SQL code for the report
Web save to My Reports	Save reports and documents to My Reports folder
Web save to Shared Reports	Save reports and documents to the Shared Reports folder
Web simple graph formatting	Perform simple graph formatting changes A user with this privilege has access to the following formatting options: • From the Format menu, all options under the Graph menu item • The Format context menu (all options) • The Graph toolbar (all options) • In the Format toolbar, the Advanced Graph Formatting option • In the Format: Graph dialog box, all options in the General tab and the Legend and Series Labels in the Format tab
Web use Object Sharing Editor	Use the Object Sharing Editor to set ACLs for objects
Web use Visual Threshold Editor	Use the Visual Threshold Editor

Web Professional privileges

These privileges correspond to the report design functionality available to users of MicroStrategy Web. The predefined MicroStrategy Web Professional group is assigned these privileges by default. The Web Professional group also inherits all of the privileges assigned to the Web Reporter and Web Analyst groups. License Manager counts any user who has any of these privileges as a Web Professional user.

MicroStrategy Web users that are licensed for MicroStrategy Report Services can view and interact with a document in Flash Mode, if the document was created before MicroStrategy 10.1 and was defined to display in Flash Mode. Certain interactions in Flash Mode have additional licensing requirements:

- Users are required to license MicroStrategy Web Analyst to pivot row or column position in a grid or cross-tabular grid of data in Flash Mode.
- Users are required to license MicroStrategy Web Professional to modify the properties of widgets used in a document in Flash Mode.

Beginning with 10.1, Presentation Mode replaces Flash Mode; newly created documents automatically display in Presentation Mode. For information on working with documents in Flash Mode and Presentation Mode, see the *MicroStrategy Web Help*.

Privilege	Allows the user to...
* Web define Intelligent Cube report	Create a report that uses an Intelligent Cube as a data source.
* Web save derived elements	Save stand-alone derived elements, separate from the report.
** Web create HTML container	Create HTML container objects in a document.
** Web document design	Create a document page, access Design Mode for documents, and perform WYSIWYG editing of documents in Flash Mode. This privilege is required to define conditional formatting.
** Web manage document datasets	Add datasets to and remove datasets from a document. The user must have the Web document design privilege as well.
**** Web configure transaction	Associate objects in a grid with a Transaction Services report.
Web define advanced report options	Set the available and default Run and Export modes for a report.
Web define MDX cube report	Define a new report that accesses an MDX cube, and see the MDX Cube option in the Create Report dialog box.
Web edit drilling and links	Use the Link Editor.
Web format grid and graph	Change the formats of grid and graph reports using the Formatting and Graph toolbars and the Formatting Panels. For documents, modify widget properties. This privilege also includes all functionality available in the Web Number Formatting and Web Simple Graph Formatting privileges (in the Web Designer privilege group).
Web modify the list of report objects (use Object Browser - all objects)	• Use the Object Browser when viewing a report in View or Design Mode. This determines whether the user is a report designer or a report creator. A report designer is a user who can build new reports based on any object in the project. A report creator can work only within the parameters of a predesigned report that has been set up by a report designer. For more information on this, see the *Advanced Reporting Guide*. • Use the All Objects panel when working with a dashboard. The All Objects panel displays the data in the current panel. For steps to use the All Objects panel, see the *MicroStrategy Web Help*.
Web save templates and filters	Save templates and filters.
Web set column widths	Modify the column widths and row height for a grid report.
Web subscribe others	View available addresses for all users, and add other MicroStrategy users to a report or document subscription.

Privilege	Allows the user to...
Web use advanced threshold editor	Use the Advanced Threshold Editor.
Web use custom group editor	Create or edit custom groups using the Custom Group Editor in Web.
Web use design mode	Modify the report using Design Mode.
Web use filter editor	Add or modify the report filter for a report.
Web use metric editor	Use the Metric Editor to create or modify metrics.
Web use prompt editor	Use the Prompt Editor to create or modify prompts.

MicroStrategy Desktop privileges

These privileges correspond to the functionality available to users of the MicroStrategy Desktop product. License Manager counts any user who has any of these privileges as a Desktop user.

Privilege	Allows the user to...
Use MicroStrategy Desktop	Connect to a server from Desktop. The Desktop user does not need any Web privileges for this server connection.

When this privilege is selected, the following privileges (in other privilege groups) are automatically selected as well, to allow the user to use functionality while connected to the server:

- **Create and edit database instances and connections**: Allows the Desktop user to access database connections while saving a dashboard to the server.
- **Create application objects**: Allows the Desktop user to access application objects while uploading a dashboard to the server.
- **Create schema objects**: Allows the Desktop user to access schema objects while uploading a dashboard to the server.
- **Export to MicroStrategy file**: Allows the Desktop user to download dashboards and datasets from the server.
- **Import MicroStrategy file**: Allows the Desktop user to upload dashboards to the server and share the dashboard with other users.
- **Use OLAP Services**: Allows the Desktop user to import datasets that use OLAP Services features from the server.
- **Web Modify Report List**: Allows the Desktop user to use a project's objects in Desktop.

- **Web publish Intelligent Cube** : Allows the Desktop user to publish an Intelligent Cube while importing datasets from the server.
- **Web run dashboard**: Allows the Desktop user to download dashboards from the server.
- **Web Use Object Sharing Editor**: Allows the Desktop user to share a dashboard by emailing a link, copying the dashboard's link URL, or automatically generating HTML code containing the link URL.

For additional information on privileges that apply to MicroStrategy Desktop and suggested best practices, see Knowledge Base Article KB241190.

Common privileges

The predefined MicroStrategy Web Reporter and Analyst user groups are assigned these common privileges by default.

Privilege	Allows the user to...
* Drill within Intelligent Cube	Drill within an Intelligent Cube, so no SQL is executed. A user who has this privilege and executes a drill that can be resolved through OLAP Services does not generate and execute SQL against the warehouse. This privilege is required for drilling in MicroStrategy Web in Flash Mode.
* Use Dynamic Sourcing	Execute reports that use Dynamic Sourcing.
* Use OLAP Services	Create and execute reports and documents that use OLAP Services features.
** Web execute document	Browse and execute Report Services documents from any XML API client, such as MicroStrategy Web or MicroStrategy Mobile.
** Web run dashboard	Execute a Visual Insight dashboard.
**** Execute transaction	Edit the grids and fields in a Transaction Services report and submit the changed results to the report.
Add notes	Add notes to a report or document.
Create application objects	Create application objects. The appropriate editor privileges listed for Developer privileges are also required.
Create new folder	Create and delete new folders.
Create schema objects	Create schema objects.
Create shortcut	Create and delete shortcuts to objects.
Edit notes	Edit an existing note in a report or document.
Export to Excel	Export a report or document to a Microsoft Excel spreadsheet.

Privilege	Allows the user to...
Export to flash	Export a report or document to a MacroMedia Flash presentation.
Export to HTML	Export a report or document to HTML.
Export to MicroStrategy file	Export a dashboard or dataset from the server.
Export to PDF	Export a report or document to a PDF document.
Export to text	Export a report or document to plain text.
Save personal prompt answer	Save personalized answers to prompts.
Schedule request	Schedule a report or document for later delivery.
Use server cache	Use the caches on Intelligence Server.
Use Translation editor	Use the Translation editor. User must have Write access to translate an object.
View History List	View the History List from any XML API client, such as MicroStrategy Web or MicroStrategy Mobile.
View notes	View notes that have been added to a report or document.

MicroStrategy Office privileges

This privilege corresponds to the functionality available to users of the MicroStrategy Office product.

Privilege	Allows the user to...
Use Office	Make requests and execute reports through the MicroStrategy Office interface

MicroStrategy Mobile privileges

These privileges correspond to the functionality available to users of the MicroStrategy Mobile product. License Manager counts any user who has this privilege as a MicroStrategy Mobile user.

Privilege	Allows the user to...
** Mobile edit dashboard	Modify a dashboard in the MicroStrategy Mobile client
** Mobile run dashboard	View a dashboard in the MicroStrategy Mobile client
** Mobile save dashboard	Save changes to a dashboard in the MicroStrategy Mobile client
** Mobile run document	View Report Services documents in the MicroStrategy Mobile client, and create Mobile document subscriptions in MicroStrategy Web

Privilege	Allows the user to...
Email screenshot from device	Email a screenshot from a mobile device
Mobile save to My Reports	Save items to the user's My Reports folder from the MicroStrategy Mobile client
Mobile save to Shared Reports	Save items to the Shared Reports folder from the MicroStrategy Mobile client
Print from device	Print a screenshot from a mobile device
Use MicroStrategy Mobile	View reports in the MicroStrategy Mobile client

Deliveries privileges

These privileges correspond to the functionality available to users of the Distribution Services product. License Manager counts any user who has any of these privileges as a Distribution Services user.

Privilege	Allows the user to...
Configure contacts data security profile (server level only)	Link and unlink a contact to a user
Create and edit transmitters and devices (server level only)	Use the Transmitter Editor and Device Editor
Create dynamic address list	Create new dynamic address lists for subscriptions
Create email address	Create new email addresses for subscriptions
Create file location	Create new file locations for subscriptions
Create FTP location	Create a new location on an FTP server for subscriptions
Create print location	Create new printer locations for subscriptions
Subscribe dynamic address list	Subscribe a dynamic address list to a report or document
Subscribe to email	Create an email subscription
Subscribe to file	Create a file subscription
Subscribe to FTP	Create an FTP subscription
Subscribe to print	Create a print subscription
Use Distribution Services	Receive email, file, FTP, and print subscriptions
Use link to History List in email	Receive an email subscription with a link to a History List, and use the Data And Link To History List and Link To History List options when creating an email subscription
Use Send Preview Now	Send a preview for email, file, FTP, print, and bulk export

Privilege	Allows the user to...
	subscriptions
Use Send Now	Send email, file, FTP, print, and bulk export subscriptions immediately

MultiSource Option privileges

These privileges correspond to the MultiSource Option functionality available in Intelligence Server.

Privilege	Allows the user to...
Execute multiple source report	Execute a report that uses a DBInstance other than the project's primary DBInstance
Import table from multiple sources	Import a table from a DBInstance other than the project's primary DBInstance

Analyst privileges

These privileges, along with the common privileges (see *Common privileges, page 323*), correspond to the basic functionality available in Developer. The predefined Analyst group is assigned these privileges by default. License Manager counts any user who has the Use Developer privilege and at least one of these privileges, but none of the Developer privileges, as an Analyst user.

Privilege	Allows the user to...
* Create derived metrics	Create calculations using metrics on a report.
* Define derived elements	Create derived elements in a report or document.
* Use Report Objects window	View and use the Report Objects window.
* Use View filter editor	Use the View filter Editor.
** Execute document	Execute documents from Developer.
Alias objects	Rename objects.
Change user preferences	Set his or her preferences.
Choose attribute form display	Set attribute forms.
Configure toolbars	Show/hide toolbars in the Report Editor.
Drill and link	Use links to view related data not shown in the original report results.

Privilege	Allows the user to...
Modify report subtotals	Change or create subtotals on a report.
Modify sorting	Change a report's sorting.
Pivot report	Pivot information on a report.
Re-execute report against warehouse	Re-execute a report, hitting the warehouse rather than the server cache. If Intelligence Server caching is turned off and this privilege is not granted, the re-execute button is removed.
Save custom autostyle	Save a report's style as a custom autostyle.
Send to e-mail	Use the Send to e-mail option in the Report Editor.
Use Data Explorer	Use the Data Explorer in the Object Browser.
Use Developer	Log in to a project using Developer. If a user does not have this privilege for a project, that project is not shown in Developer. If a user does not have this privilege for any projects in the project source, Developer returns the error message, "No projects are available for this project source."
Use Grid Options	Use the Grid Options dialog box.
Use History List	Use the History List.
Use Report Data Options	Use the Report Data Options feature.
Use Report Editor	Access the **New** option in the Report Editor, and create new reports. If a user has this privilege but not the Use design view privilege (in Developer privileges), she can still create new reports from templates, but the *Blank report* option is not available.
Use Search Editor	Use the search feature on all editors and Developer.
Use Thresholds Editor	Use the Thresholds Editor. This privilege is required to define conditional formatting.
View SQL	View a report's SQL.

Developer privileges

These privileges correspond to the report design functionality available in Developer. The predefined Developer group is assigned these privileges by default. The Developer group also inherits all the privileges assigned to the Analyst group. License Manager counts any user who has any of these privileges as a Developer user.

Privilege	Allows the user to...
* Define Intelligent Cube	Create a report that uses an Intelligent Cube as a data source.

Privilege	Allows the user to...
report	
* Publish Intelligent Cube	Publish an Intelligent Cube to Intelligence Server.
* Save derived elements	Save stand-alone derived elements, separate from the report.
* Use Intelligent Cube Editor	Create Intelligent Cubes.
** Create HTML container	Create HTML container objects in a document.
** Use Document Editor	Use the Document Editor.
*** Use bulk export editor	Use the Bulk Export Editor to define a bulk export report.
**** Define transaction report	Define a Transaction Services report using the Freeform SQL editor.
Define Freeform SQL report	Define a new report using Freeform SQL, and see the Freeform SQL icon in the Create Report dialog box.
Define MDX cube report	Define a new report that accesses an MDX cube.
Define Query Builder report	Define a new Query Builder report that accesses an external data source, and see the Query Builder icon in the Create Report dialog box.
Format graph	Modify a graph's format using a toolbar or gallery.
Modify the list of report objects (use Object Browser)	Add objects to a report, which are not currently displayed in the Report Objects window. This determines whether the user is a report designer or a report creator. A report designer is a user who can build new reports based on any object in the project. A report creator can work only within the parameters of a predesigned report that has been set up by a report designer. This privilege is required to edit the report filter and the report limit. For more information on these features, see the *Advanced Reporting Guide*.
Use Consolidation Editor	Use the Consolidation Editor.
Use Custom Group Editor	Use the Custom Group Editor.
Use Data Mart Editor	Use the Data Mart Editor.
Use design mode	Use Design View in the Report Editor.
Use Drill Map Editor	Create or modify drill maps.
Use Filter Editor	Use the Filter Editor.
Use Find and Replace dialog	Use the Find and Replace dialog box.
Use Formatting Editor	Use the formatting editor for consolidations, custom groups, and reports.
Use HTML Document Editor	Use the HTML Document Editor.
Use Link Editor	Use the Link Editor.

Privilege	Allows the user to...
Use Metric Editor	Use the Metric Editor. Among other tasks, this privilege allows the user to import DMX (Data Mining Services) predictive metrics.
Use project documentation	Use the project documentation feature to print object definitions.
Use Prompt Editor	Use the Prompt Editor.
Use SQL Statements tab in Datamart/Bulk Export editors	Use the SQL Statements tab in the Datamart Editor and the Bulk Export editor.
Use Subtotal Editor	Use the Subtotal Editor.
Use Template Editor	Use the Template Editor.
Use Translation Editor bypass	Use the Translation Editor. Users with this privilege can translate an object without having Write access to the object.
Use VLDB Property Editor	Use the VLDB Properties Editor.
View ETL information	*This privilege is deprecated.*

Architect privileges

These privileges correspond to the functionality available to users of the Architect product, that is, project designers. The predefined MicroStrategy Architect group is assigned these privileges by default. License Manager counts any user who has any of these privileges as an Architect user.

Privilege	Allows the user to...
Bypass schema objects security access checks	Modify schema objects without having the necessary permissions for each object. For example, users with this can update the schema or use the Warehouse Catalog Browser without having administrator privileges. Write permission for the folder containing the schema objects may also be required and is not granted by this privilege.
Import function	Use the Function Plug-in Editor.
Import MDX cube	Import an MDX cube.
Use Architect editors	Use the editors in Architect (for example, the Attribute, Fact, Hierarchy, and Table editors). This privilege is required to work with logical views.

Object Manager privileges

These privileges correspond to the functionality available to users of the Object Manager product. The predefined System Administrators group (a member of the System

Monitors group) is assigned these privileges by default. License Manager counts any user who has any of these privileges as an Object Manager user.

Privilege	Allows the user to...
Use Object Manager	Use Object Manager to migrate objects between projects. A user cannot log into a project source in Object Manager unless she has this privilege or Use Object Manager Read-only (below) on at least one project in that project source. Additionally, a user cannot open a project in Object Manager unless she has this privilege on that project.
Use Object Manager Read-only	Use Object Manager to view objects and create update packages but not make any changes to any projects.
Use Repository Translation Wizard	Use the Repository Translation Wizard to translate objects.

Command Manager privileges

This privilege corresponds to the functionality available to users of the Command Manager product. The predefined System Administrators group (a member of the System Monitors group) is assigned these privileges by default. License Manager counts any user who has this privileges as a Command Manager user.

These privileges cannot be granted as part of a security role.

Privilege	Allows the user to...
Use Command Manager (server level only)	Use Command Manager to run and manage scripts.

Integrity Manager privileges

This privilege corresponds to the functionality available to users of the MicroStrategy Integrity Manager product. License Manager counts any user who has this privilege as an Integrity Manager user.

Privilege	Allows the user to...
Use Integrity Manager	Use Integrity Manager to test reports in a project

Administration privileges

These privileges control access to the Administration features listed just below the project source's name. They also control access to options in the **Administration** menu.

Privilege	Allows the user to...
Administer caches	Have full control over result, element, and object caches in a project.
Administer cluster (server level only)	Add or remove nodes in a cluster, and set the default cluster membership.
Administer cubes	Publish, unpublish, delete, and update Intelligent Cubes.
Administer database connections (server level only)	Close database connections.
Administer History List	Delete other users' History List messages. This functionality is available for database-based history lists, not for file-based history lists.
Administer jobs	Kill jobs from other users.
Administer subscriptions	Create, edit, and delete subscriptions and schedule administrative tasks.
Administer user connections	Close user connections opened by other users.
Assign security filters	Grant or revoke a security filter to another user for a project.
Assign security roles	Grant or revoke a security role to another user for a project.
Audit change journal	Use the Change Journal Monitor; view the change history for all objects the user has Browse access for.
Bypass all object security access checks	Have full control over all objects regardless of the object access permissions granted to the user, and grant or revoke the Bypass All Object Security Access Checks privilege to other users. This privilege is inherently granted for use of Object Manager, Project Duplicator, and Project Mover if you have the appropriate privileges to use those tools.
Configure caches	View and set the report, document, element, and object caching properties.
Configure connection map	View, set, and refresh the connection map for a project.
Configure governing	• Project level: View and set project governing settings. • Server level: View and set governing settings for all projects and for Intelligence Server.
Configure group membership (server level only)	Add or remove users to a user group. When this privilege is granted, the Create And Edit Users And Groups privilege is automatically granted as well.
Configure history lists (server level only)	View and set the History List properties.

Privilege	Allows the user to...
Configure language settings	Use the User Language Preferences Manager.
Configure project basic	View and change project settings, and use MDUpdate utility.
Configure project data source	View and change the project's primary data source, and add and remove database instances for use in data marts, Freeform SQL, Query Builder, MDX, and data import.
Configure security settings	View and change server-level security settings.
Configure server basic (server level only)	Create, view, and change the server definition.
Configure statistics	View and change the statistics configuration.
Configure subscription settings	Set up email notifications for successful subscriptions.
Create and edit contacts and addresses (server level only)	Create and modify contacts and addresses.
Create and edit database instances and connections (server level only)	Create and modify database instances and database connections, and set the number of database threads for each database instance and the prioritization map of each database instance.
Create and edit database logins (server level only)	Create and modify database logins.
Create and edit schedules and events (server level only)	Create and modify schedules and events.
Create and edit security filters	Create and modify security filters.
Create and edit security roles (server level only)	Create and modify security roles.
Create and edit users and groups (server level only)	Create and modify users and user groups. To enable or disable users, you must have the Enable User privilege. To grant or revoke privileges, you must have the Grant/Revoke Privilege privilege.
Create configuration objects (server level only)	Create configuration objects. This privilege is inherently granted for using Object Manager if you have the Use Object Manager privilege.
Duplicate project	Use the Project Duplication Wizard to duplicate a project.
Edit project status	Use the Project Status Editor.
Enable Intelligence Server administration from Web	In Web, access the Intelligence Server Administration page.

Privilege	Allows the user to...
Enable user (server level only)	Enable or disable users. When this privilege is granted, the Create And Edit Users And Groups privilege is automatically granted as well.
Fire events (server level only)	Trigger events.
Grant/Revoke privilege (server level only)	Grant and revoke privileges to users. To grant or revoke the Bypass All Object Security Access Checks privilege to another user, you must have the Bypass All Object Security Access Checks privilege.
Idle and resume project	Idle and resume projects.
Link users and groups to external accounts (server level only)	Link MicroStrategy users and groups to users and groups from sources such as Windows NT, LDAP, or a database.
Load and unload project	Load and unload projects, and configure which projects are loaded on which cluster nodes at startup.
Monitor caches	Use the Cache Monitor; view information for all caches in a project.
Monitor cluster (server level only)	Use the Cluster view of the System Administration Monitor.
Monitor cubes	Use the Intelligent Cube Monitor; view information for all Intelligent Cubes in a project.
Monitor database connections (server level only)	Use the Database Connection Monitor; view information for all database connections in a project.
Monitor History Lists	Use the History List Monitor; view information for all history list messages in a project.
Monitor jobs	Use the Job Monitor; view information for all jobs in a project.
Monitor project	Use the Project view of the System Administration Monitor.
Monitor subscriptions	Use the Subscription Manager; view information for all subscriptions in a project. Scheduled administrative tasks are visible only if the user has the privilege corresponding to the administrative task.
Monitor user connections	Use the User Connection Monitor; view information for all user connections in a project.
Reset user password (server level only)	Change the passwords of other users.
Web administration	Access the MicroStrategy Web Administrator page and assign Project defaults.

5

COMMAND MANAGER RUNTIME

Statement Reference Guide

MicroStrategy Command Manager can perform various MicroStrategy administrative and application development tasks by means of a simple scripting language. For detailed information about Command Manager, see the *Command Manager* chapter in the *MicroStrategy System Administration Guide*.

Command Manager Runtime is a lightweight version of Command Manager for bundling with OEM applications.

See the Statement Reference Guide in Command Manager for a full list of statements and the syntax details.

Executing a script with Command Manager Runtime

Command Manager Runtime is distributed as a single executable file, `cmdmgrlt.exe`. By default, this file is installed in this directory:

```
Program Files (x86)\Common Files\MicroStrategy\
```

This file is automatically installed with any MicroStrategy installation, so you can use it to silently execute Command Manager scripts for your OEM application.

To execute a script with Command Manager Runtime, call the Command Manager Runtime executable, `cmdmgrlt.exe`, with the following parameters:

If the project source name, the input file, or an output file contain a space in the name or path, you must enclose the name in double quotes.

Effect	Parameters
Connection (required)	

Connect to a project source
 	
If `-p` is omitted, Command Manager Runtime assumes a null password.
	`-n ProjectSourceName`
	`-u UserName`
	`-p Password`

Script input (required)	
Specify the script file to be executed	`-f InputFile`
Script output (optional; choose only one)	
Log script results, error messages, and status messages to a single file	`-o OutputFile`
Log script results, error messages, and status messages to separate files, with these default file names: • `CmdMgrResults.log` • `CmdMgrFail.log` • `CmdMgrSuccess.log`	`-break`

Log script results, error messages, and status messages to separate files, with specified names
 	
You can omit one or more of these parameters. For example, if you want to log only error messages, use only the `-of` parameter.
	`-or ResultsFile`
	`-of FailFile`
	`-os SuccessFile`

Script output options (optional)	
Begin each log file with a header containing information such as the version of Command Manager used	`-h`
Print instructions in each log file and on the console	`-i`
In case of an Intelligence Server error, print the Intelligence Server error code and the Command Manager exit code in each log file and on the console	`-e`
Display script output on the console	`-showoutput`
Execution options (optional)	
Halt script execution on critical errors	`-stoponerror`

You can get the full list of parameters from a command prompt by typing `cmdmgrlt.exe /?` and pressing **Enter**.

Syntax reference guide

Command Manager Runtime uses only a subset of the commands available in the full version of Command Manager. If you try to execute a script with statements that are not available in Command Manager Runtime, the script fails with the message, "You are not licensed to run this command."

This syntax reference guide lists the statements available for Command Manager Runtime and gives the syntax and examples for each statement.

For late-breaking updates to the list of statements supported in Command Manager Runtime, see the *MicroStrategy Readme*.

Each statement in the syntax reference includes a title and four sections:

- The first section provides a brief description of the statement's function.
- The second section contains the statement's syntax, with reserved words in UPPERCASE and identifiers in *italics*.
- The third section describes identifiers and other included tokens not in a reserved-word list, and their associated restrictions.
- The fourth section provides an example statement.

Some symbols used in the syntax reference and in the outlines are not part of the syntax at all but have specific meanings related to the statement:

- Tokens between square brackets **[ ]** are optional.
- Tokens between parentheses and separated by a pipe **(|)** are exclusive options; that is, only one of the tokens is used in a statement.
- Tokens in *italics* are identifiers. They should be replaced by the desired parameter. They may have additional restrictions; see the individual statement syntax guides for these restrictions.
    - Identifiers that have an index **(1, 2, .. , n)** are of the same type. Each represents a parameter; their use may have additional restrictions.

6

INTELLIGENCE SERVER STATISTICS DATA DICTIONARY

This chapter lists the staging tables in the statistics repository to which Intelligence Server logs statistics. The detailed information includes the table name, its function, the table to which the data is moved in the Enterprise Manager repository, and the table's columns. For each column we provide the description and datatypes for DB2, MySQL, SQL Server, Oracle, Teradata, and Sybase databases. A **Bold** column name indicates that it is a primary key, and (I) indicates that the column is used in an index.

STG_CT_DEVICE_STATS

Records statistics related to the mobile client and the mobile device. This table is used when the **Mobile Clients** option is selected in the Statistics category of the Project Configuration Editor and the mobile client is configured to log statistics. The data load process moves this table's information to the CT_DEVICE_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
DAY_ID	Day the action was started.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the action was started.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the action was started.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
SERVERID	GUID of the Intelligence Server processing the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERMACHINE	Name of the Intelligence Server processing the request.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
DEVICEINSTID	Unique installation ID of the mobile app.	CHAR (40)	CHAR (40)	CHAR (40)	CHAR (40)	CHAR (40)	CHAR (40)
DEVICETYPE	Type of device the app is installed on, such as iPad, Droid, or iPhone.	VARCHAR(40)	VARCHAR2(40)	VARCHAR(40)	VARCHAR(40)	VARCHAR(40)	VARCHAR(40)
OS	Operating system of the device the app is installed on, such as iOS or Android.	VARCHAR(40)	VARCHAR2(40)	VARCHAR(40)	VARCHAR(40)	VARCHAR(40)	VARCHAR(40)
OSVER	Version of the operating system, such as 5.2.1.	VARCHAR(40)	VARCHAR2(40)	VARCHAR(40)	VARCHAR(40)	VARCHAR(40)	VARCHAR(40)
APPVER	Version of the MicroStrategy app.	VARCHAR(40)	VARCHAR2(40)	VARCHAR(40)	VARCHAR(40)	VARCHAR(40)	VARCHAR(40)
STATECOUNTER	An integer value that increments whenever the device information, such as DEVICETYPE, OS, OSVER, or APPVER,	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
	changes.						
STATECHANGETIME	Date and time when STATECOUNTER is incremented.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
RECORDTIME	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

STG_CT_EXEC_STATS

Records statistics related to execution of reports/documents in a mobile app. This table is used when the **Mobile Clients** option is selected in the Statistics category of the Project Configuration Editor and the mobile client is configured to log statistics. The data load process moves this table's information to the CT_EXEC_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the action was started.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the action was started.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the action was started.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
DEVICEINSTID (I)	Unique installation ID of the mobile app.	CHAR (40)	CHAR (40)	CHAR (40)	CHAR (40)	CHAR (40)	CHAR (40)
STATECOUNTER (I)	An integer value that	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	increments when the device information, such as DEVICETYPE, OS, OSVER, or APPVER (in STG_CT_ DEVICE_STATS), changes.						
USERID	GUID of the user making the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SESSIONID	GUID of the session that executed the request. This should be the same as the SESSIONID for this request in STG_IS_ REPORT_STATS.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CTSESSIONID	GUID of the MicroStrategy Mobile client session ID. A new client session ID is generated every time a user logs in to the mobile app.	INTEG ER	NUMBE R(10)	INTEGE R	INTEGE R	INTEG ER	INTEG ER
MESSAGEID	ID corresponding to the JOBID (in STG_IS_ REPORT_ STATS) of the message generated by the execution.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
ACTIONID	Similar to JOBID but generated by the client and cannot be NULL. The JOBID may be NULL if the user is offline during execution.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
SERVERID	GUID of the Intelligence Server processing the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERMACHINE	Name of the machine hosting the Intelligence Server processing the request.	VARCHAR (255)	VARCHAR2 (255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
REPORTID	GUID of the report used in the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
DOCUMENTID	GUID of the document used in the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
PROJECTID	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
MSERVERMACHINE	Name of the load balancing machine.	VARCHAR (255)	VARCHAR2 (255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
CTREQUESTTIME	Time when the user submits a request to the mobile app.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
CTRECEIVEDTIME	Time when the mobile app begins receiving data from MicroStrategy Mobile Server.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
CTREQRECTIME	Difference between CTRequestTime and CTReceivedTime, in milliseconds.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
CTRENDERSTARTTIME	Time when the mobile app begins rendering.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
CTRENDERFINISHTIME	Time when the mobile app finishes rendering.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
CTRENDERTIME	Difference between CTRenderStartTime and CTRenderFinishTime, in milliseconds.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
EXECUTIONTYPE	Type of report/document execution: • 1: User execution • 2: Pre-cached execution • 3: Application recovery execution • 4: Subscription cache pre-loading	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	execution • 5: Transaction subsequent action execution • 6: Report queue execution • 7: Report queue recall execution • 8: Back button execution						
CACHEIND	Whether a cache was hit during the execution, and if so, what type of cache hit occurred: • 0: No cache hit • 1: Intelligence Server cache hit • 2: Device cache hit • 6: Application memory cache hit	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
PROMPTIND	Whether the report or document is prompted: • 0: Not prompted • 1: Prompted	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
CTDATATYPE	Whether the job is for a report or a document: • 3: Report • 55: Document	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
CTNETWORKTYPE	The type of network used: • 3G • WiFi • LTE • 4G	VARCHAR (40)	VARCHAR2(40)	VARCHAR(40)	VARCHAR(40)	VARCHAR (40)	VARCHAR(40)
CTBANDWIDTH	Estimated network bandwidth, in kbps.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
VIEWFINISHTIME	Time at which the user either clicks on another report/document, or navigates away from the mobile app.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
VIEWTIME	Difference between CTRenderFinishTime and ViewFinishTime, in milliseconds.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
MANIPULATIONS	An integer value that increases with every manipulation the user makes after the report/document is rendered, excluding those that require	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	fetching more data from Intelligence Server and/or result in another report/document execution.						
CTAVGMANIPRENDERTIME	Average rendering time for each manipulation.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CTLATITUDE	Latitude of the user.	FLOAT	FLOAT	DOUBLE	FLOAT	FLOAT	FLOAT
CTLONGITUDE	Longitude of the user.	FLOAT	FLOAT	DOUBLE	FLOAT	FLOAT	FLOAT

STG_CT_MANIP_STATS

Records statistics related to manipulation of reports/documents in a mobile app. This table is used when the **Mobile Clients** and **Mobile Clients Manipulations** options are selected in the Statistics category of the Project Configuration Editor and the mobile client is configured to log statistics. The data load process moves this table's information to the CT_MANIP_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the action was started.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the action was started.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the action was started.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DEVICEINSTID (I)	Unique installation ID of the mobile app.	CHAR (40)	CHAR (40)	CHAR (40)	CHAR (40)	CHAR (40)	CHAR (40)
STATECOUNTER (I)	An integer value that increments when the device information, such as DEVICETYPE, OS, OSVER, or APPVER (in STG_CT_ DEVICE_STATS), changes.	INTEGER	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
USERID	GUID of the user making the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SESSIONID	GUID of the session that executed the request. This should be the same as the SESSIONID for this request in STG_ IS_ REPORT_STATS.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CTSESSIONID	GUID of the MicroStrategy Mobile client session ID. A new client session ID is generated every time a user logs in to the mobile app.	CHAR (32)	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
ACTIONID	Similar to JOBID but generated by the client and cannot be NULL. The JOBID may be	INTEGER	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	NULL if user is offline during execution.						
SERVERID	GUID of the Intelligence Server processing the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERMACHINE	Name of the machine hosting the Intelligence Server processing the request.	VARCHAR (255)	VARCHAR2 (255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
REPORTID	GUID of the report used in the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
DOCUMENTID	GUID of the document used in the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
PROJECTID	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
MANIPSEQUENCEID	The order in which the manipulations were made. For each manipulation, the mobile client returns a row, and the value in this column increments for each row.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
MANIPTYPEID	Type of manipulation: • 0: Unknown • 1: Selector • 2: Panel Selector • 3: Action Selector • 4: Change Layout • 5: Change View • 6: Sort • 7: Page By	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
MANIPNAME	Name of the item that was manipulated. For example, if a selector was clicked, this is the name of the selector.	VARCHAR (255)	VARCHAR2 (255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
MANIPVALUE	Value of the item that was manipulated. For example, if a panel selector was clicked, this is the name of the selected panel.	VARCHAR (2000)	VARCHAR2 (2000)	VARCHAR (2000)	VARCHAR (2000)	VARCHAR (2000)	VARCHAR (2000)
MANIPVALUESEQ	If the value for MANIPVALUE is too long to fit in a single row, this manipulation is spread over multiple rows, and this value is incremented.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
CTMANIPSTARTTIME	Time when the user submits the manipulation.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
CTMANIPFINISHTIME	Time when the mobile app finishes processing the manipulation and forwards it for rendering.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
CTMANIPTIME	Difference between CTMANIPSTARTTIME and CTMANIPFINISHTIME, in milliseconds.	FLOAT	FLOAT	DOUBLE	FLOAT	FLOAT	FLOAT
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
DETAIL1	A flexible column to capture different states of manipulation.	VARCHAR (2000)	VARCHAR2 (2000)	VARCHAR (2000)	VARCHAR (2000)	VARCHAR (2000)	VARCHAR (2000)
DETAIL2	A flexible column to capture different states of manipulation.	VARCHAR (2000)	VARCHAR2 (2000)	VARCHAR (2000)	VARCHAR (2000)	VARCHAR (2000)	VARCHAR (2000)
RECORDTIME	Date and time when this information was written to the statistics database.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

IS_ALERT_STATS

Records statistics related to Health Center alerts. This table is used when the Health Center in Configuration Wizard is configured to record alerts on performance counters in the statistics database.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID (I)	Day the action was started.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the action was started.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the action was started.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
SEQUENCE	Chronological order of a counter key, where a counter key is defined by the MACHINENAME, COUNTERCATEGORY, and COUNTERNAME values for an alert. For example, the third alert for CPU Usage in the Configuration category on Machine A has a sequence value of 3.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
ALERTTIME	Time an alert was triggered.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
MACHINENAME	Name of the machine that the alert was triggered from.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PRODUCTID	Identifier for the product that is the source of the system check.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
SYSTEMCHECKID	Identifier for the system check that was run for the alert that was triggered.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
COUNTERCATEGORY	Category of the alert, such as Configuration, MicroStrategy 3-tier Client, Intelligence Server, and so on.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
COUNTERINSTANCE	For MicroStrategy use.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
COUNTERNAME	Name of the alert, such as Memory Usage, CPU Usage, Web Server Availability, and so on.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
ALERTID	Identifier for the alert that was triggered.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
ALERTDESC	The text of the alert.	VARCHAR (2000)	VARCHAR2(255)	VARCHAR(2000)	VARCHAR(2000)	VARCHAR (2000)	VARCHAR (2000)
READFLAG	Indicates whether the alert has been seen by an administrator.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
ACTEDFLAG	Indicates whether the alert has been acted on by an administrator.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
EMAILADDRESSES	Email addresses to which the alert was sent.	VARCHAR (2000)	VARCHAR2 (2000)	VARCHAR(2000)	VARCHAR(2000)	VARCHAR (2000)	VARCHAR (2000)
ERRORMESSAGE	The error message displayed to the user when an error is encountered.	VARCHAR (2000)	VARCHAR2 (2000)	VARCHAR(2000)	VARCHAR(2000)	VARCHAR (2000)	VARCHAR (2000)

STG_IS_CACHE_HIT_STATS

Tracks job executions that hit the report cache. This table is used when the **Basic Statistics** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_CACHE_HIT_ STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the job execution hit the report cache.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the job execution hit the report cache.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the job execution hit the report cache.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
CACHEINDEX (I)	A sequential number for this table.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
CACHESESSIONID (I)	GUID of the user session.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERID	GUID of the server definition.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CACHEHITTIME (I)	Timestamp when this cache is hit.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
CACHEHITTYPE (I)	Type of cache hit: • 0: Report cache hit • 1 or 2: Document cache hit	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
CACHECREATORJOBID (I)	Job ID that created the cache.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
CREATORSESSIONID (I)	GUID for the session in which cache was created.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
JOBID (I)	Job ID for partial cache hit, or document parent job ID if the cache hit originated from document child report.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
STARTTIME	Timestamp of when the job started.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
RECORDTIME (I)	Timestamp of when the record was written to the database, according to	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	database system time.						
SERVERMACHINE	(Server machine name:port number) pair.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PROJECTID (I)	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

The table below lists combinations of CACHEHITTYPE and JOBID that can occur in the STG_IS_CACHE_HIT_STATS table and what those combinations mean.

Cache Hit Type	JobID	Description
0	-1	For a normal report, a full cache hit
0	Real JobID	For a normal report, a partial cache hit
1	Parent JobID	For a child report from a document, a full cache hit, so no child report
2	Child JobID	For a child report from a document, a partial cache hit, child report has a job

STG_IS_CUBE_REP_STATS

Records statistics related to Intelligent Cube manipulations. This table is not populated unless at least one of the **Advanced Statistics Collection Options** are selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_CUBE_REP_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
DAY_ID (I)	Day the action was started.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the action was started.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the action was started.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
SESSIONID	GUID of the session that executed the action on the Intelligent Cube.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
JOBID	Job ID for the action on the Intelligent Cube.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
PROJECTID	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
STARTTIME	Timestamp of when the action started.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
FINISHTIME	Timestamp of when the action finished.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
CUBEREPORTGUID	GUID of the Intelligent Cube report that was executed.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CUBEINSTANCEID	GUID of the Intelligent Cube instance in memory.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CUBEACTIONID	Type of action against the Intelligent Cube:	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
	• 0: Reserved for MicroStrategy use • 1: Cube Publish • 2: Cube View Hit • 3: Cube Dynamic Source Hit • 4: Cube Append • 5: Cube Update • 6: Cube Delete • 7: Cube Destroy						
REPORTGUID	If a report hit the Intelligent Cube, the GUID of that report.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CUBEKBSIZE	If the Intelligent Cube is published or refreshed, the size of the Intelligent Cube in KB.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
CUBEROWSIZE	If the Intelligent Cube is published or refreshed, the number of rows in the Intelligent Cube.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
SERVERMACHINE	Name of the Intelligence Server	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
	processing the request.						
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
RECORDTIME	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

STG_IS_DOC_STEP_STATS

Tracks each step in the document execution process. This table is used when the **Document Job Steps** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_DOC_STEP_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the document was requested for execution.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the document was requested for execution.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the document was requested for execution.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
JOBID	GUID of the document job.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
STEPSEQUENCE	Sequence number for a job's steps.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
SESSIONID	GUID of the user session.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERID	GUID of the server definition.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
STEPTYPE	Type of step. For a description, see *Report and document steps, page 399*. • 1: Metadata object request step • 2: Close job • 3: SQL generation • 4: SQL execution • 5: Analytical Engine server task • 6: Resolution server task • 7: Report net server task • 8: Element request step • 9: Get report instance • 10: Error message send task • 11: Output	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	message send task • 12: Find report cache task • 13: Document execution step • 14: Document send step • 15: Update report cache task • 16: Request execute step • 17: Data mart execute step • 18: Document data preparation • 19: Document formatting • 20: Document manipulation • 21: Apply view						

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	context • 22: Export engine • 23: Find Intelligent Cube task • 24: Update Intelligent Cube task • 25: Post-processing task • 26: Delivery task • 27: Persist result task • 28: Document dataset execution task						
STARTTIME	Timestamp of the step's start time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
FINISHTIME	Timestamp of the step's finish time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
QUEUETIME	Time duration, in milliseconds, between the last step finish and the next step start.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
CPUTIME	CPU time, in milliseconds, used during this step.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
STEPDURATION	FINISHTIME minus STARTTIME, in milliseconds.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
RECORDTIME	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
SERVERMACHINE	(Server machine name:port number) pair.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PROJECTID	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

STG_IS_DOCUMENT_STATS

Tracks document executions that the Intelligence Server processes. This table is used when the **Basic Statistics** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_DOCUMENT_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the document was requested for	DATE	TIMESTAMP	DATE	DATE	DATE	DATE

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	execution.						
HOUR_ID	Hour the document was requested for execution.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the document was requested for execution.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
JOBID (I)	Job ID.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
SESSIONID (I)	GUID of the user session.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERID	GUID of the Intelligence Server's server definition at the time of the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERMACHINE	Server machine name or IP address.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PROJECTID	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
USERID	GUID of theuser.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
DOCUMENTID (I)	GUID of the document.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
REQUESTRECTIME	The timestamp at which the request is received.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
REQUESTQUEUETIME	Total queue time of all steps in this request.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
STARTTIME	Time duration between request receive time and document job was created. An offset of the RequestRecTime.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
FINISHTIME	Time duration between request receive time and document job last step was finished. An offset of the RequestRecTime.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
EXECERRORCODE	Execution error code. If no error, the value is 0.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
REPORTCOUNT	Number of reports included in the document.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
CANCELINDICATOR	Was the document job canceled?	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
PROMPTINDICATOR	Number of prompts in the report.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
CACHEDINDICATOR	Was the document cached?	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
RECORDTIME (I)	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
CPUTIME	CPU time, in milliseconds, used for document execution.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
STEPCOUNT	Total number of steps involved in execution (not just unique steps).	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
EXECDURATION	Duration of execution, in milliseconds.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
ERRORMESSAGE	Error message displayed to the user when an error is encountered.	VARCHAR (4000)	VARCHAR2 (4000)	VARCHAR(4000)	VARCHAR(4000)	VARCHAR (4000)	VARCHAR (4000)
EXECACTIONS	Intelligence Server-related actions that need to take place during document execution.	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
EXECFLAGS	Intelligence Server-related processes needed to refine the document execution.	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER	INTEGER
PROMPTANSTIME	Total time, in milliseconds, the user spent answering prompts on the document.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
EXPORTINDC	1 if the document was exported, otherwise 0.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
CACHECREATORJOBID	If the job hit a cache, the job ID of the job that created the cache used by the current job.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
CACHECREATORSESSONID	If the job hit a cache, the GUID for the session in which the cache was created.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
MESSAGEID	For MicroStrategy use.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

STG_IS_INBOX_ACT_STATS

Records statistics related to History List manipulations. This table is used when the **Inbox Messages** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_INBOX_ACT_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID (I)	Day the manipulation was started.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the manipulation was started.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the manipulation was started.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
SESSIONID (I)	GUID of the session that started the History List manipulation.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERID	GUID of the server definition of the Intelligence Server being manipulated.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERMACHINE	Name and port number of the Intelligence Server machine where the manipulation is taking place.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PROJECTID	GUID of the project where the History List message is mapped.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
INBOXACTION	Type of manipulation: • 0: Reserved for MicroStrategy use • 1: Add: Add message to History List • 2: Remove: Remove message from History List • 3: Rename: Rename message • 4: Execute: Execute contents of message • 5: Change Status: Change message status from Ready to Read • 6: Requested: Retrieve message contents • 7: Batch Remove: Intelligen	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	ce Server bulk operation, such as cache expiration						
USERID	ID of the user doing the manipulation.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
OWNERID	ID of the user that created the message.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
MESSAGEID	GUID of the History List message being acted on.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
MESSAGETITLE	Name of the report or document referenced in the History List message.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
MESSAGEDISPNAME	User-defined name of the History List message. Blank unless the user has renamed the History List message.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
CREATIONTIME	Date and time when the History List message was created.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
STARTTIME	Date and time when the manipulation started.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
REPORTJOBID (I)	Report job ID for the History List Message Content Request. Blank if no job was executed or if a document was executed.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
DOCUMENTJOBID (I)	Document job ID for the History List Message Content Request. Blank if no job was executed or if a report was executed.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
SUBSCRIPTIONID	ID of the subscription that invoked the manipulation.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
ACTIONCOMMENT	• If the manipulation is a batch deletion of History List messages, this field contains the condition or SQL statement used to delete the messages. • If there is an error,	VARCHAR (4000)	VARCHAR2 (4000)	VARCHAR(4000)	VARCHAR(4000)	VARCHAR (4000)	VARCHAR (4000)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	this field holds the error message.						
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
RECORDTIME	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

STG_IS_MESSAGE_STATS

Records statistics related to sending messages through Distribution Services. This table is used when the **Subscriptions** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_MESSAGE_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the job was requested for execution.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the job was requested for execution.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the job was requested for execution.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
MESSAGEINDEX	Message GUID used to identify a message.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
SESSIONID	GUID of the user session created to generate the message.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
HISTORYLISTMESSAGEID	History List message ID. If there is no History List message associated with the subscription, this value is 00000000 00000000 00000000 00000000.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SCHEDULEJOBID	Job ID of report/document executed to run the subscription instance. If no job is created, this value is -1. If a fresh job A is created and it hits the cache of an old job B, SCHEDULEJOBID takes the value of the fresh job A.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DATATYPE	Type of subscribed object: • 3: Report • 55: Document	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
RECIPIENTCONTACTID	GUID of the message recipient.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
DELIVERYTYPE	Type of subscription: • 1: Email • 2: File • 4: Printer • 8: Custom • 16: History List • 32: Client • 40: Cache update • 128: Mobile • 100: Last one • 255: All	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
SUBSINSTID	Subscription instance GUID used to send the message.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
SCHEDULEID	Schedule GUID. If there is no schedule associated with the subscription, this value is -1.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SUBINSTNAME	Name of the subscription.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
DATAID	GUID of the data content.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CONTACTTYPE	The contact type for this subscription instance's RecipientID.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
RECIPIENTGROUPID	Recipient's group ID for group messages sent to a Contact Collection or a User Group.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
RECIPIENTCONTACTNAME	Name of the contact who received the message.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
ISDEFAULTADDRESS	Whether the address that the message was sent to is the default address of a MicroStrategy user: • 0: No • 1: Yes	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
ADDRESSID	GUID of the	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	address the message was sent to.						
DEVICEID	ID of the device the message was sent to.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
ISNOTIFICATIONMESSAGE	Whether a notification was sent: • 0: No • 1: Yes	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
NOTIFICATIONADDR	Address ID the notification is sent to.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
SERVERID	Server definition GUID under which the subscription ran.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERMACHINE	Server machine name or IP address under which the report or document job ran.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PROJECTID	Project GUID under which the data content resides.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
EXECSTARTTIME	Time at which the message creation started.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
EXECFINISH TIME	Time at which the message delivery	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	finished.						
DELIVERYSTATUS	Status of the message delivery.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
PHYSICALADDRESS	Email address the message was sent to.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
BATCHID		CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
RECORDTIME	Timestamp of when the record was written to the table.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

STG_IS_PERF_MON_STATS

Records statistics related to notification, diagnostics, and performance counters logged by Intelligence Server. This table is used when the performance counters in the Diagnostics and Performance Monitoring Tool are configured to record statistics information. The data load process moves this table's information to the IS_PERF_MON_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
DAY_ID (I)	Day the performance counter was recorded.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the performance counter was recorded.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the	SMALLI	NUMBER	SMALLINT	SMALLINT	SMALLI	SMALLI

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
	performance counter was recorded.	NT	(5)			NT	NT
SERVER_MACHINE	The server machine that logs the notification message.	VARCHAR(255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
COUNTER_CAT	The category of the counter, such as Memory, MicroStrategy Server Jobs, or MicroStrategy Server Users.	VARCHAR(255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
COUNTER_INSTANCE	For MicroStrategy use.	VARCHAR(255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
COUNTER_NAME	Name of the performance counter.	VARCHAR(255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
EVENT_TIME	Timestamp of when the event occurred in Intelligence Server.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
COUNTER_VALUE	Counter value.	FLOAT	FLOAT	DOUBLE	FLOAT	FLOAT	FLOAT
CTR_VAL_TYP	Counter value type.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
PROJECTID	GUID of the project.	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
RECORDTIME	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

STG_IS_PR_ANS_STATS

Records statistics related to prompts and prompt answers. This table is used when the **Prompts** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_PR_ANS_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
DAY_ID	Day the prompt was answered.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the prompt was answered.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the prompt was answered.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
JOBID	Job ID assigned by the server.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
SESSIONID	GUID of the user session.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
PR_ ORDER_ID	Order in which prompts were answered. Prompt order is set in Developer's Prompt Ordering dialog box.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
ANS_SEQ_ ID	Sequence ID. For MicroStrategy use.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
PR_LOC_TYPE	The COM object type of the object that the prompt resides in: • 1: Filter • 2: Template • 12: Attribute	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
PR_LOC_ID	ID of the object that the prompt resides in.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)
PR_LOC_DESC	Object name of the object that the prompt resides in.	VARCHAR(255)	VARCHAR2(255)	VARCHAR (255)	VARCHAR (255)	VARCHAR(255)	VARCHAR(255)
PR_GUID	GUID of the prompt.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)
PR_NAME	Name of the prompt.	VARCHAR(255)	VARCHAR2(255)	VARCHAR (255)	VARCHAR (255)	VARCHAR(255)	VARCHAR(255)

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
PR_TITLE	Prompt title. This cannot be NULL. This is the text that is displayed in Developer's Prompt Ordering dialog box, under Title.	VARCHAR(255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
PR_ANS_TYPE	Type of prompt. For example, element, expression, object, or numeric.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
PR_ANSWERS	Prompt answers.	VARCHAR(4000)	VARCHAR2(4000 CHAR)	VARCHAR(4000)	VARCHAR(4000)	VARCHAR(4000)	VARCHAR(4000)
PR_ANS_GUID	For MicroStrategy use.	VARCHAR(4000)	VARCHAR2(4000)	VARCHAR(4000)	VARCHAR(4000)	VARCHAR(4000)	VARCHAR(4000)
IS_REQUIRED	Y: If a prompt answer is required. N: If a prompt answer is not required.	CHAR	CHAR	CHAR	CHAR	CHAR	CHAR
SERVERID	GUID of the server definition.	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)
PROJECTID	GUID of the project.	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)

Column	Description	SQL Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
SERVERMACHINE	The Intelligence Server machine name and IP address.	VARCHAR(255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
STARTTIME	Timestamp of the job start time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
RECORDTIME	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
REPOSITORYID	GUID of the metadata repository.	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)

STG_IS_PROJ_SESS_STATS

Records statistics related to project session. This table is used when the **Basic Statistics** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_PROJ_SESS_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the project session was started.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the project session was started.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
MINUTE_ID	Minute the project session was started.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
SESSIONID	Session object GUID. This is the same session ID used in STG_IS_SESSION_STATS.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	If you close and reopen the project connection without logging out from Intelligence Server, the session ID is reused.						
SERVERID	Server definition GUID.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
SERVERMACHINE	The Intelligence Server machine name and IP address.	VARCHAR(255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
USERID	GUID of the user performing the action.	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)
PROJECTID	Project GUID.	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)
CONNECTTIME	Timestamp of when the session was opened.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
DISCONNECTTIME (I)	Timestamp of when the session was closed.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
RECORDTIME (I)	Timestamp of when the record was written to the statistics database.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
REPOSITORYID	GUID of the metadata repository.	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)

STG_IS_REP_COL_STATS

Tracks the column-table combinations used in the SQL during report executions. This table is used when the **Report job tables/columns accessed** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves

this table's information to the IS_REP_COL_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the report was requested for execution.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the report was requested for execution.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the report was requested for execution.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
JOBID	Report job ID.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
SESSIONID	GUID of the user session.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (23)	CHAR (23)
SERVERID	GUID of the server definition.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
TABLEID	GUID of the database tables used.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
COLUMNID	GUID of the columns used.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
COLUMNNAME	Description of the column used.	VARCHAR (255)	VARCHAR2 (255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
SQLCLAUSETYPEID	The SQL clause in which the column is being used.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
COUNTER	The number of times a specific column/table/clause type combination occurs within a report execution.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
STARTTIME	Timestamp of the job start time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
RECORDTIME	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
SERVERMACHINE	(Server machine name:port number) pair.	VARCHAR (255)	VARCHAR2 (255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PROJECTID	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

STG_IS_REP_SEC_STATS

Tracks executions that used security filters. This table is used when the **Basic Statistics** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_REP_SEC_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the job was requested for execution.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
HOUR_ID	Hour the job was requested for execution.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the job was requested for execution.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
JOBID (I)	Job ID.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
SESSIONID (I)	GUID of the user session.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SECURITYFILTERSEQ	Sequence number of the security filter, when multiple security filters are used.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
SERVERID	Server definition GUID.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SECURITYFILTERID (I)	Security filter GUID.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
STARTTIME	Timestamp of when the job started.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
RECORDTIME	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
SERVERMACHINE	(Server machine name:port number) pair.	VARCHAR(255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
PROJECTID	GUID of the project.	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)
REPOSITORYID	GUID of the metadata repository.	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR(32)

STG_IS_REP_SQL_STATS

Enables access to the SQL for a report execution. This table is used when the **Report Job SQL** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_REP_SQL_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the SQL pass was started.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
HOUR_ID	Hour the SQL pass was started.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the SQL pass was started.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
JOBID	Job ID.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
SQLPASSSEQUENCE	Sequence number of the SQL pass.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
SESSIONID	GUID of the user session.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERID	GUID of the server definition.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
STARTTIME	Start timestamp of the SQL pass.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
FINISHTIME	Finish timestamp of the SQL pass.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
EXECTIME	Execution time, in milliseconds, for the SQL pass.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
SQLSTATEMENT	SQL used in the pass.	VARCHAR (4000)	VARCHAR2 (4000)	VARCHAR(4000)	VARCHAR(4000)	VARCHAR (4000)	VARCHAR (4000)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
SQLPASSTYPE	Type of SQL pass: • 0: SQL unknown • 1: SQL select • 2: SQL insert • 3: SQL create • 4: Analytical • 5: Select into • 6: Insert into values • 7: Homogen. partition query • 8: Heterogen. portend query • 9: Metadata portend pre-query • 10: Metadata portend list pre-query • 11: Empty	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	• 12: Create index • 13: Metric qual. break by • 14: Metric qual. threshold • 15: Metric qual. • 16: User-defined • 17: Homogen. portend loop • 18: Homogen. portend one tbl • 19: Heterogen. portend loop • 20: Heterogen. portend one tbl • 21: Insert fixed values into • 22:						

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	Datamart from Analytical Engine • 23: Clean up temp resources • 24: Return elm number • 25: Incremental elem browsing • 26: MDX query • 27: SAP BI • 28: Intelligent Cube instruc • 29: Heterogen. data access • 30: Excel file data import • 31: Text file data import • 32: Database table import						

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	• 33: SQL data import						
TOTALTABLEACCESSED	Number of tables hit by the SQL pass.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
DBERRORMESSAGE	Error message returned from database.	VARCHAR (4000)	VARCHAR2 (4000)	VARCHAR(4000)	VARCHAR(4000)	VARCHAR (4000)	VARCHAR (4000)
RECORDTIME	Timestamp of when the record was written to the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
SERVERMACHINE	(Server machine name:port number) pair.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PROJECTID	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
DBINSTANCEID	GUID of the physical database instance.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
DBCONNECTIONID	GUID of the database connection.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
DBLOGINID	GUID of the database login.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SQLSTATEMENTSEQ	Sequence number of the SQL statement.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

STG_IS_REP_STEP_STATS

Tracks each step in the report execution process. This table is used when the **Report Job Steps** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_REP_STEP_ STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the report was requested for execution.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the report was requested for execution.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the report was requested for execution.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
JOBID	Job ID.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
STEPSEQUENCE	Sequence number for a job's steps.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
SESSIONID	GUID of the user session.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERID	GUID of the Intelligence Server processing the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
STEPTYPE	Type of step. For a description, see *Report and document steps, page 399*. • 1: Metadata object request step • 2: Close job • 3: SQL generation • 4: SQL execution • 5: Analytical Engine server task • 6: Resolution server task • 7: Report net server task • 8: Element request step • 9: Get report instance • 10: Error message send task • 11: Output	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	message send task • 12: Find report cache task • 13: Document execution step • 14: Document send step • 15: Update report cache task • 16: Request execute step • 17: Data mart execute step • 18: Document data preparation • 19: Document formatting • 20: Document manipulation • 21: Apply view						

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	context • 22: Export engine • 23: Find Intelligent Cube task • 24: Update Intelligent Cube task • 25: Post-processing task • 26: Delivery task • 27: Persist result task • 28: Document dataset execution task						
STARTTIME	Timestamp of the step's start time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
FINISHTIME	Timestamp of the step's finish time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
QUEUETIME	Time duration between last step finish and the next step start, in milliseconds.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
CPUTIME	CPU time used during this step, in milliseconds.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
STEPDURATION	FINISHTIME - STARTTIME, in milliseconds	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
RECORDTIME	Timestamp of when the record was logged in the database, according to database system time	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
SERVERMACHINE	(Server machine name:port number) pair.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PROJECTID	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Report and document steps

This IS_REP_STEP_TYPE table lists the Intelligence Server tasks involved in executing a report or a document. These are the possible values in the STEPTYPE column in the IS_REP_STEP_STATS and IS_DOC_STEP_STATS tables.

Task name	Task description
1: MD Object Request	Requesting an object definition from the project metadata.
2: Close Job	Closing a job and removing it from the list of pending jobs.
3: SQL Generation	Generating SQL that is required to retrieve data, based on schema.
4: SQL Execution	Executing SQL that was generated for the report.
5: Analytical Engine	Applying analytical processing to the data retrieved from the data source.

Task name	Task description
6: Resolution Server	Loading the definition of an object.
7: Report Net Server	Transmitting the results of a report.
8: Element Request	Browsing attribute elements.
9: Get Report Instance	Retrieving a report instance from the metadata.
10: Error Message Send	Sending an error message.
11: Output Message Send	Sending a message other than an error message.
12: Find Report Cache	Searching or waiting for a report cache.
13: Document Execution	Executing a document
14: Document Send	Transmitting a document
15: Update Report Cache	Updating report caches
16: Request Execute	Requesting the execution of a report
17: Data Mart Execute	Executing a data mart report
18: Document Data Preparation	Constructing a document structure using data from the document's datasets
19: Document Formatting	Exporting a document to the requested format
20: Document Manipulation	Applying a user's changes to a document
21: Apply View Context	Reserved for MicroStrategy use
22: Export Engine	Exporting a document or report to PDF, plain text, Excel spreadsheet, or XML
23: Find Intelligent Cube	Locating the cube instance from the Intelligent Cube Manager, when a subset report, or a standard report that uses dynamic caching, is executed.
24: Update Intelligent Cube	Updating the cube instance from the Intelligent Cube Manager, when republishing or refreshing a cube.
25: Post-processing	Reserved for MicroStrategy use.
26: Delivery	Used by Distribution Services, for email, file, or printer deliveries of subscribed-to reports/documents.
27: Persist Result	Persists execution results, including History List and other condition checks. All subscriptions hit this step, although only subscriptions that persist results (such as History List) perform actions in this step.
28: Document Dataset Execution	Waiting for child dataset reports in a document to execute.

STG_IS_REPORT_STATS

Tracks job-level statistics information about every report that Intelligence Server executes to completion. This table is used when the **Basic Statistics** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_REPORT_STATS table, which has the same columns and datatypes.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the report was requested for execution.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the report was requested for execution.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the report was requested for execution.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
JOBID (I)	Job ID.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
SESSIONID (I)	GUID of the user session.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERID	GUID of the Intelligence Server's server definition that made the request.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERMACHINE	Server machine name, or IP address if the machine name is not available.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
PROJECTID	GUID of the project.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
USERID	GUID of the user.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
REPORTID	GUID of the report. For an ad hoc report, the Template ID is created on the fly and there is no corresponding object with this GUID in the object lookup table.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
FILTERID	GUID of the filter.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
EMBEDDEDFILTER	1 if an embedded filter was used in the report, otherwise 0.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
TEMPLATEID	GUID of the template. For an ad hoc report, the Template ID is created on the fly, and there is no corresponding object with this GUID in the object lookup table.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
EMBEDDEDTEMPLATE	1 if an embedded template was used in the report, otherwise 0.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
PARENTJOBID (I)	Job ID of the parent document job, if the current job is a document	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	job's child. -1 if the current job is not a document job's child.						
DBINSTANCEID	GUID for the physical database instance.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
DBUSERID	Database user ID for the physical database instance.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
PARENTINDICATOR	1 if this job is a document job's child, otherwise 0.	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
REQUESTRECTIME	Timestamp when the request is received.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
REQUESTQUEUETIME	Total queue time of all steps in this request.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
EXECSTARTTIME	Time passed before the first step started. An offset of the RequestRecTime.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
EXECFINISHTIME	Time passed when the last step is finished. An offset of the RequestRecTime.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
SQLPASSES	Number of SQL passes.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	INTEGER
JOBERRORCODE	Job error code. If no error, the value is 0.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
CANCELINDICATOR	1 if the job was canceled, otherwise 0.	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
ADHOCINDICATOR	1 if the report was ad hoc, otherwise 0. This includes any executed job that is not saved in the project as a report (for example: drilling results, attribute element prompts, creating and running a report before saving it).	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
PROMPTINDICATOR	Number of prompts in the report.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
DATAMARTINDICATOR	1 if the report created a data mart, otherwise 0.	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
ELEMENTLOADINDIC	1 if the report was a result of an element browse, otherwise 0.	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DRILLINDICATOR	1 if the report was the result of a drill, otherwise 0.	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
SCHEDULEINDICATOR	1 if the report was run from a schedule, otherwise 0.	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
CACHECREATEINDIC	1 if the report created a cache, otherwise 0.	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT(1)
PRIORITYNUMBER	Query execution step priority.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
JOBCOST	User-supplied report cost.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
FINALRESULTSIZE	Number of rows in the report.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
RECORDTIME (I)	Timestamp of when the record was logged in the database, according to the database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
ERRORMESSAGE	The error message displayed to the user when an error is encountered.	VARCHAR (4000)	VARCHAR2 (4000)	VARCHAR(4000)	VARCHAR(4000)	VARCHAR (4000)	VARCHAR (4000)
DRILLTEMPLATEUNIT	For MicroStrategy use. GUID of the object that was drilled from.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
NEWOBJECT	For MicroStrategy	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	use. GUID of the object that was drilled to.						
DRILLTYPE	For MicroStrategy use. Enumeration of the type of drilling action performed.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
TOTALTABLEACCESS	Total number of unique tables accessed by the report during execution.	SMALLINT	NUMBER(5)	SMALLINT	SMALLINT	SMALLINT	INTEGER
SQLLENGTH	Length in characters of the SQL statement. For multiple passes, this value is the sum of SQL statement lengths of each pass.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
EXECDURATION	Duration of the report execution, in milliseconds.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
CPUTIME	CPU time used for report execution, in milliseconds.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
STEPCOUNT	Total number of steps involved in execution (not just unique steps).	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
EXECACTIONS	Intelligence Server-related actions that need to take place during report execution.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
EXECFLAGS	Intelligence Server-related processes needed to refine the report execution.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
DBERRORINDIC	1 if a database error occurred during execution, otherwise 0.	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT
PROMPTANSTIME	Total time, in milliseconds, the user spent answering prompts on the report.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER
CUBEINSTANCEID	GUID of the Intelligent Cube used in a Cube Publish or Cube Hit job.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CUBESIZE	Size, in KB, of the Intelligent Cube in memory for a Cube Publish job.	INTEGER	NUMBER(10)	INTEGER	INTEGER	INTEGER	INTEGER

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
SQLEXECINDIC	1 if any SQL was executed against the database, otherwise 0.	BIT	NUMBER(1)	SMALLINT	BYTEINT	BIT	TINYINT
EXPORTINDIC	1 if the report was exported, otherwise 0.	TINYINT	NUMBER(3)	SMALLINT	BYTEINT	TINYINT	TINYINT
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
MESSAGEID	ID of the message.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

STG_IS_SCHEDULE_STATS

Tracks which reports have been run as the result of a subscription. This table is used when the **Subscriptions** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_SCHEDULE_STATS table, which has the same columns and datatypes.

Column	Description	SQLU Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
DAY_ID (I)	Day the job was requested for execution.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID (I)	Hour the job was requested for execution.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID (I)	Minute the job was requested for execution.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT

Column	Description	SQLU Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
SCHEDULE ID (I)	Job ID.	INTEGER	NUMBER (10)	INTEGER	INTEGER	INTEGER	INTEGER
SESSIONID (I)	GUID of the user session.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)
SERVERID	GUID for server definition.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)
TRIGGERID (I)	GUID of the object that triggered the subscription.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)
SCHEDULETYPE (I)	Type of schedule: 0 if it is a report, 1 if it is a document	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
HITCACHE	0 if the job does not hit the cache, 1 if it does.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
STARTTIME	Timestamp of the schedule start time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
RECORDTIME (I)	Timestamp of when the record was logged in the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

Column	Description	SQLU Server Datatype	Oracle Datatype	DB2 Datatype	Teradata Datatype	Sybase Datatype	MySQL Datatype
SERVERMACHINE	(Server machine name:port number) pair.	VARCHAR(255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
PROJECTID (I)	GUID of the project.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR(32)	CHAR(32)	CHAR(32)	CHAR (32)	CHAR (32)

STG_IS_SESSION_STATS

Logs every Intelligence Server user session. This table is used when the **Basic Statistics** option is selected in the Statistics category of the Project Configuration Editor. The data load process moves this table's information to the IS_SESSION_STATS table, which has the same columns and datatypes.

The STG_IS_SESSION_STATS table does not contain project-level information and is therefore not affected by statistics purges at the project level. For details about statistics purges, see the *System Administration Guide*.

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DAY_ID	Day the session was started.	DATE	TIMESTAMP	DATE	DATE	DATE	DATE
HOUR_ID	Hour the session was started.	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT
MINUTE_ID	Minute the session was started.	SMALLINT	NUMBER (5)	SMALLINT	SMALLINT	SMALLINT	SMALLINT
SESSIONID (I)	GUID of the user session.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
SERVERID	Server definition GUID.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
SERVERMACHINE	(Server machine name:port number) pair.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
USERID	GUID of the user.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)
CLIENTMACHINE	Client machine name, or IP address if the machine name is not available.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
EVENTSOURCE	Source from which the session originated: • 1: Developer • 2: Intelligence Server Administrator • 3: Web Administrator • 4: Intelligence Server • 5: Project Upgrade • 6: Web • 7: Scheduler • 8: Custom App • 9: Narrowcast Server • 10: Object Mgr	TINYINT	NUMBER (3)	SMALLINT	BYTEINT	TINYINT	TINYINT

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	• 11: ODBO Provider • 12: ODBO Cube Designer • 13: Command Mgr • 14: Enterprise Mgr • 15: Command Line Interface • 16: Project Builder • 17: Config Wiz • 18: MD Scan • 19: Cache Utility • 20: Fire Event • 21: MicroStrategy Java admin clients • 22: MicroStrategy Web Services • 23: MicroStrategy Office • 24: MicroStrategy tools • 25: Portal Server						

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
	• 26: Integrity Mgr • 27: Metadata Update • 28: Reserved for MicroStrategy use • 29: Scheduler for Mobile • 30: Repository Translation Tool • 31: Health Center • 32: Cube Advisor • 33: Operations Mgr						
RECORDTIME (I)	Timestamp of when the record was logged in the database, according to database system time.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
WEBMACHINE	Web server machine from which a web session originates.	VARCHAR (255)	VARCHAR2(255)	VARCHAR(255)	VARCHAR(255)	VARCHAR (255)	VARCHAR (255)
CONNECTTIME (I)	Timestamp of when the session is opened.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME

Column	Description	SQL Server Data-type	Oracle Data-type	DB2 Data-type	Teradata Data-type	Sybase Data-type	MySQL Data-type
DISCONNECTTIME	Timestamp of when the session is closed.	DATETIME	TIMESTAMP	TIMESTAMP	TIMESTAMP	DATETIME	DATETIME
REPOSITORYID	GUID of the metadata repository.	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)	CHAR (32)

STG_MSI_STATS_PROP

For MicroStrategy use. Provides information about the statistics database properties. Intelligence Server uses this table to initialize statistics logging.

Column Name	Column Description
PROP_NAME	Property name, such as statistics database version, upgrade script, and so on.
PROP_VAL	Property value, such as statistics database version number, time that an upgrade script was run, and so on.

7

ENTERPRISE MANAGER DATA MODEL AND OBJECT DEFINITIONS

Detailed information about Enterprise Manager objects is available from within MicroStrategy:

- To view details about Enterprise Manager objects, walk through the Project Documentation wizard in the Enterprise Manager project. To access this, from MicroStrategy Developer, open the Enterprise Manager project, click the **Tools** menu and select **Project Documentation**.
- To view details about Enterprise Manager schema objects, such as facts, attributes, and hierarchies, open the Enterprise Manager project in MicroStrategy Architect. To access this, from MicroStrategy Developer, open the Enterprise Manager project, click the **Schema** menu and select **Architect**.
- For information about configuring Enterprise Manager and how you can use it to help tune the MicroStrategy system and information about setting up project documentation so it is available to networked users, see the Enterprise Manager Guide .

Enterprise Manager data warehouse tables

The following is a list of tables in the Enterprise Manager data warehouse.

Temporary tables are created and used by the data loading process when data is migrated from the statistics tables to the Enterprise Manager warehouse. These temporary tables are the following:

- IS_REP_SQL_TMP
- IS_REP_STEP_TMP
- IS_SESSION_TMP1
- IS_PROJECT_FACT_1_TMP
- EM_IS_LAST_UPD_1
- EM_IS_LAST_UPD_2

Fact tables

- *CT_EXEC_FACT*
- *CT_MANIP_FACT*
- *IS_CONFIG_PARAM_FACT*
- *IS_CUBE_ACTION_FACT*
- *IS_DOC_FACT*
- *IS_DOC_STEP_FACT*
- *IS_INBOX_ACT_FACT*
- *IS_MESSAGE_FACT*
- *IS_PERF_MON_FACT*
- *IS_PR_ANS_FACT*
- *IS_PROJECT_FACT_1*
- *IS_REP_COL_FACT*
- *IS_REP_FACT*
- *IS_REP_SEC_FACT*
- *IS_REP_SQL_FACT*
- *IS_REP_STEP_FACT*
- *IS_SESSION_FACT*
- *IS_SESSION_MONITOR*

CT_EXEC_FACT

Contains information about MicroStrategy Mobile devices and report/document executions and manipulations. Created as a view based on columns in the source tables listed below.

Source tables

- CT_DEVICE_STATS: Statistics table containing information about the mobile client and the mobile device
- CT_EXEC_STATS: Statistics table containing information about mobile report and document execution
- IS_SERVER: Lookup table that provides descriptive information about the server definitions being tracked
- IS_REP: Lookup table that provides descriptive information about the reports being tracked
- IS_DOC: Lookup table that provides descriptive information about the documents being tracked
- IS_PROJ: Lookup table that provides descriptive information about the projects being tracked
- EM_MD: Lookup table for metadata
- EM_USER: Lookup table for users

List of table columns

Column Name	Column Description
CT_DEVICE_INST_ID	Unique installation ID of the mobile app.
CT_STATE_COUNTER	An integer value that increments when the device information, such as DEVICETYPE, OS, OSVER, or APPVER (in CT_DEVICE_STATS), changes.
CT_STATE_CHANGE_TS	Date and time when STATECOUNTER is incremented.
CT_DEVICE_TYPE	Type of device the app is installed on, such as iPad 2, Droid, or iPhone 6.
CT_OS	Operating system the app is installed on, such as iOS or Android.
CT_OS_VER	Version of the operating system, such as 5.2.1.
CT_APP_VER	Version of the MicroStrategy app.
EM_USER_ID	ID of the user executing the document.
IS_SESSION_ID	GUID of the session that executed the request. This should be the same as the SESSIONID for this request in IS_REP_FACT.

Column Name	Column Description
CT_SESSION_ID	GUID of the MicroStrategy Mobile client session ID. A new client session ID is generated every time a user logs in to the mobile app.
IS_MESSAGE_ID	ID corresponding to the JOBID (in IS_REP_FACT) of the message generated by the execution.
CT_ACTION_ID	Similar to JOBID but generated by the client and cannot be NULL. JOBID may be NULL if the user is offline during execution.
IS_SERVER_ID	GUID of the Intelligence Server processing the request.
EM_APP_SRV_MACHINE	Name and port number of the Intelligence Server machine where the mobile document execution is taking place.
IS_REP_ID	GUID of the report used in the request.
IS_DOC_ID	GUID of the document used in the request.
IS_PROJ_ID	GUID of the project.
IS_REPOSITORY_ID	GUID of the metadata repository.
EM_MOB_SRV_ MACHINE	Name and port number of the Mobile Server machine where the mobile document execution is taking place.
CT_REQ_TS	Time when the user submits a request to the mobile app.
CT_REC_TS	Time when the mobile app begins receiving data from MicroStrategy Mobile Server.
CT_REQ_REC_TM_MS	Difference in milliseconds between CT_REQ_TS and CT_REC_TS.
CT_RENDER_ST_TS	Time when the mobile app begins rendering.
CT_RENDER_FN_TS	Time when the mobile app finishes rendering.
CT_RENDER_TM_MS	Difference in milliseconds between CT_RENDER_ST_TS and CT_RENDER_ FN_TS
CT_EXEC_TYPE_IND_ID	Type of report or document execution: • 1: User execution • 2: Pre-cached execution • 3: System recovery execution • 4: Subscription cache pre-loading execution • 5: Transaction subsequent action execution • 6: Report queue execution • 7: Report queue recall execution • 8: Back button execution
CT_CACHE_HIT_IND_ID	Whether a cache was hit during the execution, and if so, what type of cache hit occurred:

Column Name	Column Description
	• 0: No cache hit • 1: Intelligence Server cache hit • 2: Device cache hit • 6: Application memory cache hit
CT_PROMPT_IND_ID	Whether the report or document is prompted: • 0: Not prompted • 1: Prompted
CT_DATATYPE_ID	Whether the job is for a report or a document: • 3: Report • 55: Document
CT_NETWORK_TYPE	The type of network used: • 3G • WiFi • LTE • 4G
CT_BANDWIDTH_KBPS	Estimated network bandwidth, in kbps.
CT_VIEW_FN_TS	Time at which the user either clicks on another report or document, or backgrounds the mobile app.
CT_VIEW_TM_MS	Difference in milliseconds between CT_RENDER_FN_TS and CT_VIEW_FN_TS.
CT_NU_OF_MANIP	An integer value that increases with every manipulation the user makes after the report or document is rendered, excluding those that require fetching more data from Intelligence Server or that result in another report or document execution.
CT_AVG_MANIP_RENDER_TM_MS	Average rendering time for each manipulation.
CT_LATITUDE	Latitude of the user.
CT_LONGITUDE	Longitude of the user.
DAY_ID	Day the action was started.
CT_TIMESTAMP	Time the manipulation was started.
HOUR_ID	Hour the action was started.
MINUTE_ID	Minute the action was started.
EM_RECORD_TS	Date and time when this information was written to the statistics database.

Column Name	Column Description
CT_REQ_RECEIVED_ FLAG	Whether the manipulation request was received.
CT_REQ_RENDERED_ FLAG	Whether the manipulation was completed.
CT_REQ_HAS_DEVICE_ FLAG	Whether the manipulation request was made by a mobile app.
CT_JOB_ID	The ID of the job requesting the manipulation. A combination of the IS_ SESSION_ID, CT_SESSION_ID, and CT_ACTION_ID.
IS_DOC_NAME	Name of the document used in the request, or if it is a deleted document.
IS_PROJ_NAME	Name of the project used for the request or if it is a deleted project.
EM_USER_NAME	Name of the user who made the request or if it is a deleted user.
EM_LDAPLINK	Name of the user in the LDAP system or if it is a deleted user.
EM_NTLINK	Name of the user in Windows or if it is a deleted user.

CT_MANIP_FACT

Contains information about MicroStrategy Mobile devices and report/document manipulations. Created as a view based on columns in the source tables listed below.

Source tables

- CT_MANIP_STATS: Statistics table containing information about the report or document manipulations
- EM_MD: Lookup table for metadata
- IS_PROJ: Lookup table that provides descriptive information about the projects being tracked
- IS_DOC: Lookup table that provides descriptive information about the documents being tracked
- IS_REP: Lookup table that provides descriptive information about the reports being tracked
- EM_USER: Lookup table for users

List of table columns

Column Name	Column Description
CT_JOB_ID	The ID of the job requesting the manipulation. A combination of the IS_SESSION_ID, CT_SESSION_ID, and CT_ACTION_ID.
CT_DEVICE_INST_ID	Unique installation ID of the mobile app.
CT_STATE_COUNTER	An integer value that increments when the device information, such as DEVICETYPE, OS, OSVER, or APPVER (in CT_MANIP_STATS), changes.
EM_USER_ID	ID of the user making the request.
IS_SESSION_ID	GUID of the session that executed the request.
CT_SESSION_ID	GUID of the MicroStrategy Mobile client session ID. A new client session ID is generated every time a user logs in to the mobile app.
CT_ACTION_ID	Similar to JOBID but generated by the client and cannot be NULL. JOBID may be NULL if the user is offline during execution.
EM_APP_SRV_MACHINE	Name and port number of the Intelligence Server machine where the manipulation is taking place.
IS_REP_ID	GUID of the report used in the request.
IS_DOC_ID	Integer ID of the document that was executed.
IS_PROJ_ID	Integer ID of the project.
IS_MANIP_SEQ_ID	The order in which the manipulations were made in a session. For each manipulation, the mobile client returns a row, and the value in this column increments for each row.
IS_MANIP_TYPE_ID	Type of manipulation: • 0: Unknown • 1: Selector • 2: Panel Selector • 3: Action Selector • 4: Change Layout • 5: Change View • 6: Sort • 7: Page By
IS_MANIP_NAME	Name of the item that was manipulated. For example, if a selector was clicked, this is the name of the selector.
IS_MANIP_VALUE	Value of the item that was manipulated. For example, if a panel selector was clicked, this is the name of the selected panel.
IS_MANIP_VALUE_SEQ	If the value for IS_MANIP_VALUE is too long to fit in one row, this manipulation is spread over multiple rows, and this value is incremented.

Column Name	Column Description
DETAIL1	A flexible column to capture different states of manipulation.
DETAILS2	A flexible column to capture different states of manipulation.
CT_MANIP_ST_TS	Time when the user submitted the manipulation.
CT_MANIP_FN_TS	Time when the mobile app finished processing the manipulation and forwarded it for rendering.
CT_MANIP_TM_MS	Difference between CT_MANIP_ST_TS and CT_MANIP_FN_TS, in milliseconds.
DAY_ID	Day the manipulation was started.
HOUR_ID	Hour the manipulation was started.
MINUTE_ID	Minute the manipulation was started.
EM_RECORD_TS	Date and time when this information was written to the statistics database.
REP_ID	ID of the report used in the request.

IS_CONFIG_PARAM_FACT

Contains information about Intelligence Server and project configuration settings.

Related lookup tables

- IS_CONFIG_PARAM: Lookup table for configuration settings
- IS_PROJ: Lookup table for projects
- IS_SERVER: Lookup table for Intelligence Server definitions

List of table columns

Column Name	Column Description
IS_CONFIG_TS	Timestamp when the configuration setting was recorded.
IS_MD_ID	Integer ID of the metadata being monitored.
IS_PROJ_ID	ID of the project recording the configuration setting. If the configuration setting is an Intelligence Server setting, this value is 0.
IS_SERVER_ID	Integer ID of the Intelligence Server definition.
IS_CONFIG_PARAM_ID	Integer ID the configuration setting.
IS_CONFIG_PARAM_ VALUE	Value of the configuration setting.

IS_CUBE_ACTION_FACT

Contains information about Intelligent Cube manipulations. Created as a view based on columns in the source tables listed below.

Source tables

- EM_MD: Lookup table for metadata
- IS_CUBE_REP_STATS: Statistics table containing information about Intelligent Cube manipulations
- IS_CUBE_ACTION_TYPE: Lookup table listing the manipulations that can occur
- IS_PROJ: Lookup table for projects
- IS_REP: Lookup table for report objects

List of table columns

Column Name	Column Description
DAY_ID	Day the action was started.
HOUR_ID	Hour the action was started.
MINUTE_ID	Minute the action was started.
EM_RECORD_TS	Date and time when this information was written to the statistics database.
IS_SESSION_ID	GUID of the session that started the action against the Intelligent Cube.
IS_REP_JOB_ID	Job ID for the action on the Intelligent Cube
IS_PROJ_ID	Integer ID of the project where the Intelligent Cube is stored.
IS_CUBE_REP_ID	Integer ID of the Intelligent Cube report that was published, if any
IS_CUBE_INST_ID	GUID of the Intelligent Cube instance in memory

Column Name	Column Description
IS_CUBE_ACT_ID	Type of action against the Intelligent Cube: • 0: Reserved for MicroStrategy use • 1: Cube Publish • 2: Cube View Hit • 3: Cube Dynamic Source Hit • 4: Cube Append • 5: Cube Update • 6: Cube Delete • 7: Cube Destroy
IS_REP_ID	Integer ID of the report that hit the Intelligent Cube, if any.
IS_CUBE_SIZE_KB	If the Intelligent Cube is published or refreshed, size of the Intelligent Cube in KB.
IS_CUBE_ROWS	If the Intelligent Cube is published or refreshed, number of rows in the Intelligent Cube.
IS_REPOSITORY_ID	Integer ID of the metadata repository.

IS_DOC_FACT

Contains information on the execution of a document job.

Primary key:

- DAY_ID2
- IS_SESSION_ID
- IS_DOC_JOB_SES_ID
- IS_DOC_JOB_ID
- IS_DOC_CACHE_IDX

Source tables

- IS_DOCUMENT_STATS: Statistics table containing information about document executions
- EM_IS_LAST_UPD_2: Configuration table that drives the loading process (for example, data loading window)

Related lookup tables

- EM_USER: Lookup table for users

- IS_DOC: Lookup table for documents
- IS_SESSION: Lookup table for session objects

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp of when the information was recorded by Intelligence Server into the IS_DOCUMENT_STATS table.
EM_LOAD_TS	Timestamp of when the Enterprise Manager data load process began.
IS_SERVER_ID	Integer ID of the server where the session was created.
IS_SESSION_ID	GUID of the current session object.
IS_DOC_JOB_SES_ID	GUID of the session that created the cache if a cache was hit in this execution; otherwise, current session (default behavior).
IS_DOC_JOB_ID	Integer ID of the document job execution.
IS_DOC_CACHE_IDX	Always 0; not yet available, documents not currently cached. Integer ID of the cache hit index; similar to Job ID but only for cache hits. -1 if no cache hit.
IS_CACHE_HIT_ID	Always 0; not yet available, documents not currently cached. Indicates whether the job hit a cache.
IS_CACHE_CREATE_ID	Always 0, not yet available. Indicates whether a cache was created.
IS_PROJ_ID	Integer ID of the project logged into.
IS_CUBE_EXEC_ST_TS	Date and time when cube execution was started by Intelligence Server.
IS_CUBE_EXEC_FN_TS	Date and time when cube execution was finished by Intelligence Server.
EM_USER_ID	Integer ID of the user who created the session.
IS_DOC_ID	Integer ID of the document that was executed.
IS_DOC_REQ_TS	Timestamp of the execution request; request of the current session.
IS_DOC_EXEC_REQ_TS	Timestamp of the execution request; request time of the original execution request if a cache was hit, otherwise current session's request time.
IS_DOC_EXEC_ST_TS	Timestamp of the execution start.
IS_DOC_EXEC_FN_TS	Timestamp of the execution finish.
IS_DOC_QU_TM_MS	Queue duration in milliseconds.
IS_DOC_CPU_TM_MS	CPU duration in milliseconds.
IS_DOC_EXEC_TM_MS	Execution duration in milliseconds.
IS_DOC_NBR_REPORTS	Number of reports contained in the document job execution.

Column Name	Column Description
IS_DOC_NBR_PU_STPS	Number of steps processed in the document job execution.
IS_DOC_NBR_PROMPTS	Number of prompts in the document job execution.
IS_JOB_ERROR_ID	Integer ID of the job's error message, if any.
IS_CANCELLED_ID	Indicates whether the job was cancelled.
DAY_ID2	Integer ID of the day. Format YYYYMMDD.
HOUR_ID	Integer ID of the hour. Format HH (24 hours).
MINUTE_ID2	Integer ID of the minute. Format HHMM (24 hours)
DAY_ID	*This column is deprecated.*
MINUTE_ID	*This column is deprecated.*

IS_DOC_STEP_FACT

Contains information on each processing step of a document execution. Created as a view based on columns in the source tables listed below.

Source tables

- IS_DOC_STEP_STATS: Statistics table containing information about processing steps of document execution
- IS_PROJ: Lookup table for projects
- IS_DOCUMENT_STATS: Statistics table containing information about document executions
- IS_SESSION: Lookup table for session objects

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp of when the information was recorded by Intelligence Server into the _STATS tables.
IS_PROJ_ID	Integer ID of the project logged into.
IS_DOC_JOB_SES_ID	GUID of the session that created the cache if a cache was hit in this execution; otherwise, current session (default behavior).
IS_DOC_JOB_ID	Integer ID of the document job execution.

Column Name	Column Description
IS_DOC_STEP_SEQ_ID	Integer ID of the document job execution step.
IS_DOC_STEP_TYP_ID	Integer ID of the document job execution step type.
IS_DOC_EXEC_ST_TS	Timestamp of the execution start.
IS_DOC_EXEC_FN_TS	Timestamp of the execution finish.
IS_DOC_QU_TM_MS	Queue duration in milliseconds.
IS_DOC_CPU_TM_MS	CPU duration in milliseconds.
IS_DOC_EXEC_TM_MS	Execution duration in milliseconds.
DAY_ID	Day the job was executed.
HOUR_ID	Hour the job was executed.
MINUTE_ID	Minute the job was executed.

IS_INBOX_ACT_FACT

Contains information about History List manipulations. Created as a view based on columns in the source tables listed below.

Source tables

- IS_INBOX_ACT_STATS: Statistics table containing information about History List manipulations
- IS_INBOX_ACTION: Lookup table listing the manipulations that can occur

List of table columns

Column Name	Column Description
DAY_ID	Day the manipulation was started.
HOUR_ID	Hour the manipulation was started.
MINUTE_ID	Minute the manipulation was started.
IS_SESSION_ID	GUID of the session that started the History List manipulation.
IS_SERVER_ID	GUID of the server definition of the Intelligence Server being manipulated.
EM_APP_SRV_MACHINE	Name and port number of the Intelligence Server machine where the manipulation is taking place.
IS_PROJ_ID	GUID of the project where the History List message is mapped.

Column Name	Column Description
IS_INBOX_ACTION_ID	Type of manipulation: • 0: Reserved for MicroStrategy use. • 1: Add: Add message to History List • 2: Remove: Remove message from History List • 3: Rename: Rename message • 4: Execute: Execute contents of message • 5: Change Status: Change message status from Ready to Read • 6: Requested: Retrieve message contents • 7: Batch Remove: Intelligence Server bulk operation, such as cache expiration
EM_USER_ID	ID of the user doing the manipulation.
IS_HL_MESSAGE_ID	GUID of the History List message being acted on.
IS_HL_MESSAGE_TITLE	Name of the report or document referenced in the History List message.
IS_HL_MESSAGE_DISP	User-defined name of the History List message. Blank unless the user has renamed the History List message.
IS_CREATION_TS	Date and time when the History List message was created.
IS_ACT_START_TS	Date and time when the manipulation started.
IS_REP_JOB_ID	Report job ID for the History List Message Content Request. Blank if no job was executed or if a document was executed.
IS_DOC_JOB_ID	Document job ID for the History List Message Content Request. Blank if no job was executed or if a report was executed.
IS_SUBSCRIPTION_ID	ID of the subscription that invoked the manipulation
IS_ACTION_COMMENT	• If the manipulation is a batch deletion of History List messages, this field contains the condition or SQL statement used to delete the messages. • If there is an error, this field holds the error message.
EM_RECORD_TS	Date and time when this information was written to the statistics database.

IS_MESSAGE_FACT

Records all messages sent through Distribution Services.

Source table

- IS_MESSAGE_STATS: Statistics table containing information about sent messages

Related lookup tables

- IS_SCHED: Lookup table for schedules
- IS_PROJ: Lookup table for projects
- IS_SERVER: Lookup table for Intelligence Server definitions
- IS_DEVICE: Lookup table for devices
- EM_MD: Lookup table for metadata

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp of when information was recorded by Intelligence Server into the IS_MESSAGE_STATS table.
EM_LOAD_TS	Timestamp of when the Enterprise Manager data load process began.
IS_MESSAGE_INDEX	Reserved for MicroStrategy use.
IS_SESSION_ID	GUID of the session object.
DAY_ID	Integer ID of the day. Format: YYYYMMDD.
HOUR_ID	Integer ID of the hour. Format HH (24 hours).
MINUTE_ID	Integer ID of the minute.
IS_HL_MESSAGE_ID	Message ID of the job created.
IS_SCHEDULE_JOB_ID	Job ID from Intelligence Server for the subscription job.
IS_DATATYPE_ID	Type of data generated for the subscription. • 3: Report • 55: Document
IS_RCPT_CONTACT_ID	GUID of the user who is receiving the data.

Column Name	Column Description
IS_DELIVERY_TYPE_ID	Type of delivery: • 1: Email • 2: File • 4: Printer • 8: Custom • 16: History List • 20: Client • 40: Cache • 100: *(MicroStrategy use only)* • 128: Mobile • 255: *(MicroStrategy use only)*
IS_SUBS_INST_ID	GUID of the subscription.
IS_SUBS_INST_NAME	Name of the subscription.
IS_SCHEDULE_ID	GUID of the schedule that triggered the subscription, or -1 if not applicable.
IS_DATA_ID	GUID of the report or document requested.
IS_CONTACT_TYPE_ID	Type of contact delivered to: • 1: Contact • 2: Contact group • 4: MicroStrategy user • 5: Count • 8: MicroStrategy user group • 10: LDAP user • 31: *(MicroStrategy use only)*
IS_RCPT_GROUP_ID	GUID of the group receiving the subscription, or NULL if no group.
IS_RCPT_CONTACT_NAME	Name of the contact recipient.
IS_DFLT_ADDR	Indicates whether the address where the content was delivered is the default.
IS_ADDRESS_ID	GUID of the address delivered to.
IS_DEVICE_ID	ID of the Distribution Services device used in the delivery.
IS_NOTIF_MSG	Indicates whether a delivery notification message is sent.
IS_NOTIF_ADDR	GUID of the notification address.

Column Name	Column Description
IS_SERVER_ID	Numeric ID of the server definition.
IS_PROJ_ID	Numeric ID of the source project.
IS_EXEC_ST_TM_TS	Start time for subscription execution.
IS_EXEC_FM_TM_TS	Finish time for subscription execution.
IS_DELIVERY_STATUS	Indicates whether the delivery was successful.
IS_PHYSICAL_ADD	Physical address for delivery.
IS_BATCH_ID	Reserved for MicroStrategy use.
EM_APP_SRV_MACHINE	Name of the Intelligence Server.

IS_PERF_MON_FACT

Contains information about job performance .

Source table

- IS_PERF_MON_STATS: Statistics table containing information about job performance

Related lookup table

- IS_PROJ: Lookup table for projects

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp of when the information was recorded by Intelligence Server into the _STATS table.
EM_LOAD_TS	Timestamp of when the Enterprise Manager data load process began.
EM_APP_SRV_ MACHINE	The name of the Intelligence Server machine logging the statistics.
IS_COUNTER_CAT	The category of the counter, such as Memory, MicroStrategy Server Jobs, or MicroStrategy Server Users.
IS_COUNTER_ INSTANCE	MicroStrategy use.
IS_COUNTER_NAME	The name of the performance counter.

Column Name	Column Description
IS_EVENT_TIME	Timestamp of when the event occurred in Intelligence Server.
IS_COUNTER_VALUE	The value of the performance counter.
IS_CTR_VAL_TYP	The type of performance counter.
IS_PROJ_ID	Integer ID of the project logged into.
DAY_ID	Integer ID of the day. Format YYYMMDD.
HOUR_ID	Integer ID of the hour. Format HH (24 hours).
MINUTE_ID	Integer ID of the minute.

IS_PR_ANS_FACT

Contains information about prompt answers. Created as a view based on columns in the source tables listed below.

Source tables

- EM_MD: Lookup table for metadata
- EM_PR_ANS_TYPE: Lookup table for prompt answer type
- IS_PR_ANS_STATS: Statistics table containing information about session activity
- IS_PROJ: Lookup table for projects
- IS_PROMPT: Lookup table for prompts
- IS_SERVER: Lookup table for Intelligence Server definitions
- LU_OBJ_TYPE: Lookup table for COM object type

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp when the information was recorded by Intelligence Server into the _STATS table.
IS_REP_JOB_ID	Job ID assigned by the server.
IS_SESSION_ID	GUID for the user session.
PR_ORDER_ID	Order in which prompts were answered.
PR_ANS_SEQ	Sequence ID. For MicroStrategy use.

Column Name	Column Description
PR_LOC_ID	ID of the object that the prompt resides in.
PR_LOC_TYPE	COM object type of the object that the prompt resides in.
PR_LOC_DESC	Object name of the object that the prompt resides in.
PR_ANS_GUID	Reserved for MicroStrategy use.
PR_ANSWERS	Prompt answers.
PR_ANS_TYPE	Prompt answer type.
IS_SERVER_ID	Integer ID of the server where the session was created.
PR_ID	Integer ID of the prompt.
PR_GUID	GUID of the prompt.
PR_TITLE	Prompt title.
PR_NAME	Prompt name.
IS_REQUIRED	Y if a prompt answer is required, N if a prompt answer is not required.
IS_PROJ_ID	Integer ID of the project logged into.
IS_PROJ_NAME	Project name.
EM_APP_SRV_ MACHINE	The Intelligence Server machine name and IP address.
DAY_ID	Day the prompt was answered.
HOUR_ID	Hour the prompt was answered.
MINUTE_ID	Minute the prompt was answered.
IS_REPOSITORY_ID	Integer ID of the metadata repository.

IS_PROJECT_FACT_1

Represents the number of logins to a project in a day by user session and project.

Source tables

- IS_PROJ_SESSION_STATS: Statistics table containing information on session activity by project
- IS_SESSION_STATS: Statistics table containing information about session activity on Intelligence Server
- IS_SERVER: Lookup table for Intelligence Server definitions
- EM_USER: Lookup table for users
- IS_PROJ: Lookup table for projects

List of table columns

Column Name	Column Description
IS_SESSION_ID	GUID of the session object.
IS_PROJ_ID	Integer ID of the project logged into.
IS_SERVER_ID	Integer ID of the server where the session was created.
EM_APP_SRV_ MACHINE	The name of the Intelligence Server machine logging the statistics.
EM_USER_ID	Integer ID of the user who created the session.
IS_CONNECT_TS	Timestamp of the beginning of the session (login).
IS_DISCONNECT_TS	Timestamp of the end of the session (logout). NULL if the session is still open at the time of Enterprise Manager data load.
IS_TMP_DISCON_TS	Represents temporary end of a session, if that session is still open. Used to calculate the session time.
IS_SESSION_TM_SEC	Duration within the hour, in seconds, of the session.
EM_RECORD_TS	Timestamp when the information was recorded by Intelligence Server into the _STATS table.
EM_LOAD_TS	Timestamp of when the Enterprise Manager data load process began.
DAY_ID	Integer ID of the day. Format YYYMMDD.
HOUR_ID	Hour the user logged in.
MINUTE_ID	Minute the user logged in.
IS_REPOSITORY_ID	Integer ID of the metadata repository.

IS_REP_COL_FACT

Used to analyze which data warehouse tables and columns are accessed by MicroStrategy report jobs, by which SQL clause they are accessed (SELECT, FROM, and so on), and how frequently they are accessed. This fact table is at the level of a Report Job rather than at the level of each SQL pass executed to satisfy a report job request. The information available in this table can be useful for database tuning. Created as a view based on columns in the source tables listed below.

Source tables

- IS_REP_COL_STATS: Statistics table containing information about column-table combinations used in the SQL during report executions
- IS_SESSION: Lookup table for session objects
- IS_REP_FACT: Fact table for report job executions
- IS_DB_TAB: Lookup table for database tables

- IS_COL: Lookup table for columns

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp when information was recorded by Intelligence Server into the _STATS tables.
IS_JOB_ID	Integer ID of the report job execution.
IS_SESSION_ID	GUID of the current session object.
IS_COL_GUID	GUID of the column object.
IS_TABLE_ID	Integer ID of the physical database table that was used.
IS_COL_NAME	Name of the column in the database table that was used.
SQL_CLAUSE_TYPE_ID	Integer ID of the type of SQL clause (SELECT, FROM, WHERE, and so on).
COUNTER	The number of times a specific column/table/clause type combination occurs within a report execution.
DAY_ID	Day the job was executed.
HOUR_ID	Hour the job was executed.
MINUTE_ID	Minute the job was executed.

IS_REP_FACT

Contains information about report job executions.

Primary key:

- DAY_ID2
- IS_SESSION_ID
- IS_REP_JOB_SES_ID
- IS_REP_JOB_ID
- IS_DOC_JOB_ID
- IS_REP_CACHE_IDX

Source tables

- IS_CACHE_HIT_STATS: Statistics table containing information about job executions that hit a cache
- IS_DOC_FACT: Fact table containing information about document job executions

- IS_DOCUMENT_STATS: Statistics table containing information about document job executions
- IS_REP_SEC_STATS: Statistics table containing information about job executions with security filters
- IS_REPORT_STATS: Statistics table containing information about report job executions
- IS_SCHEDULE_STATS: Statistics table containing information about job executions run by a schedule
- EM_IS_LAST_UPD_2: Configuration table that drives the loading process (for example, data loading window)

Related lookup tables

- IS_SESSION: Lookup table for session objects
- IS_REP: Lookup table for report objects
- IS_TEMP: Lookup table for template objects
- IS_FILT: Lookup table for filter objects
- IS_SCHED: Lookup table for schedule objects
- IS_DOC: Lookup table for document objects

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp when the information was recorded by Intelligence Server into the _STATS table.
EM_LOAD_TS	Timestamp of when the Enterprise Manager data load process began.
IS_SERVER_ID	Integer ID of the server where the session was created.
IS_SESSION_ID	GUID of the current session object.
IS_REP_JOB_SES_ID	GUID of the session that created the cache if a cache was hit in this execution; otherwise, current session (default behavior).
IS_REP_JOB_ID	Integer ID of the report job execution.
IS_REP_CACHE_IDX	Integer ID of the cache hit index; similar to Job ID but only for cache hits. -1 if no cache hit.
IS_CACHE_HIT_ID	Indicates whether the job hit a cache.
IS_CACHE_CREATE_ID	Indicates whether a cache was created.

Column Name	Column Description
EM_USER_ID	Integer ID of the user who created the session.
EM_DB_USER_ID	DB User used to log in to the warehouse.
IS_DB_INST_ID	Integer ID of the db instance object.
IS_PROJ_ID	Integer ID of the project logged in to.
IS_REP_ID	Integer ID of the report object.
IS_EMB_FILT_IND_ID	Indicates whether the report filter is embedded.
IS_EMB_TEMP_IND_ID	Indicates whether the report template is embedded.
IS_FILT_ID	Integer ID of the filter object.
IS_TEMP_ID	Integer ID of the template object.
IS_DOC_JOB_ID	• Integer ID of the parent document execution if current report is a child of a document. • Integer ID of the parent document execution of the original report if a cache was hit. • Otherwise, -1.
IS_DOC_ID	Integer ID of the parent document object if current report is a child of a document. Otherwise, -1.
IS_REP_REQ_TS	Timestamp of the execution request; request of the current session.
IS_REP_EXEC_REQ_TS	Timestamp of the execution request; request time of the original execution request if a cache was hit, otherwise current session's request time.
IS_REP_EXEC_ST_TS	Timestamp of the execution start.
IS_REP_EXEC_FN_TS	Timestamp of the execution finish.
IS_REP_QU_TM_MS	Queue duration in milliseconds.
IS_REP_CPU_TM_MS	CPU duration in milliseconds.
IS_REP_EXEC_TM_MS	Execution duration in milliseconds.
IS_REP_ELAPS_TM_MS	Difference between start time and finish time; includes time for prompt responses.
IS_REP_NBR_SQL_PAS	Number of SQL passes.
IS_REP_RESULT_SIZE	Number of rows in the result set.
IS_REP_SQL_LENGTH	Not yet available. Number of characters.
IS_REP_NBR_TABLES	Not yet available. Number of tables.
IS_REP_NBR_PU_STPS	Number of steps processed in the execution.

Column Name	Column Description
IS_REP_NBR_PROMPTS	Number of prompts in the report execution.
IS_JOB_ERROR_ID	Integer ID of the job's error message, if any.
IS_ERROR_IND_ID	Indicates whether the job got an error.
IS_DB_ERROR_IND_ID	Indicates whether the database returned an error.
IS_CANCELLED_ID	Indicates whether the job was canceled.
IS_AD_HOC_ID	Indicates whether the job was created ad hoc.
IS_DATAMART_ID	Indicates whether the job created a data mart.
IS_ELEM_LOAD_ID	Indicates whether the job was the result of an element load.
IS_DRILL_ID	Indicates whether the job was the result of a drill.
IS_SEC_FILT_IND_ID	Indicates whether the job had a security filter associated with it.
IS_SEC_FILT_ID	Integer ID of the security filter applied.
IS_SCHED_ID	Integer ID of the schedule that executed the job.
IS_SCHED_IND_ID	Indicates whether the job was executed by a schedule.
IS_REP_PRIO_NBR	Priority of the report execution.
IS_REP_COST_NBR	Cost of the report execution.
DAY_ID2	Integer ID of the day. Format YYYYMMDD.
HOUR_ID	Integer ID of the hour. Format HH (24 hours).
MINUTE_ID2	Integer ID of the minute. Format HHMM (24 hours).
DRILLFROM	Integer ID of an attribute, metric, or other object that is drilled from.
DRILLFROM_OT_ID	Integer ID for the object type of the object that is drilled from.
DRILLTO	Integer ID of an attribute, template, or other object that is drilled to.
DRILLTO_OT_ID	Integer ID for the object type of the object that is drilled to.
DRILLTYPE	Integer flag indicating the type of drill performed (for example, drill to template, drill to attribute, and so on).
ERRORMESSAGE	Error message returned by Intelligence Server.
IS_CACHE_SESSION_ID	Alphanumeric ID of the session that created the cache on Intelligence Server.
IS_CACHE_JOB_ID	Integer ID of the job that created the cache on Intelligence Server.
IS_REP_PMT_ANS_TS	Data and time when the prompt was answered.
IS_SQL_EXEC_IND_ID	Integer ID indicating if this job hit generated SQL and hit a database or not.

Column Name	Column Description
IS_EXPORT_IND_ID	Integer ID indicating if this was an export job or not.
IS_CUBE_INST_ID	GUID of the Intelligent Cube object (if job hits it).
IS_CUBE_SIZE	Size of the Intelligent Cube the job hits (if applicable).
IS_REP_PR_ANS_TM_MS	Time in milliseconds of how long the user took to answer the prompt.
IS_EXEC_FLAG	Internal flag that indicates the type of job execution.
IS_REPOSITORY_ID	Integer ID of the metadata repository.
IS_MESSAGE_ID	Internal alphanumeric ID attached to every job.
DAY_ID	*This column is deprecated.*
MINUTE_ID	*This column is deprecated.*

IS_REP_SEC_FACT

Contains information about security filters applied to report jobs. Created as a view based on columns in the source tables listed below.

Source tables

- IS_REP_FACT: Contains information about report job executions
- IS_REP_SEC_STATS: Statistics table containing information about job executions with security filters
- IS_SEC_FILT: Provides descriptive information about the security filters being tracked
- IS_SF_ATT: Relationship table between security filters and attributes

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp when the information was recorded by Intelligence Server into the _STATS table.
EM_LOAD_TS	Timestamp of when the Enterprise Manager data load process began.
IS_PROJ_ID	Integer ID of the project logged in to.
IS_REP_JOB_SES_ID	GUID of the session that created the cache if a cache was hit in this execution; otherwise, current session (default behavior).

Column Name	Column Description
IS_REP_JOB_ID	Integer ID of the report job execution.
IS_REP_SEC_FILT_ID	Integer ID of the security filter.
IS_ATT_ID	Integer ID of the attribute.
DAY_ID	Day the job was requested for execution.
HOUR_ID	Hour the job was requested for execution.
MINUTE_ID	Minute the job was requested for execution.
IS_REPOSITORY_ID	Integer ID of the metadata repository.

IS_REP_SQL_FACT

Contains the SQL that is executed on the warehouse by report job executions. Created as a view based on columns in the source tables listed below.

Source tables

- IS_REP_FACT: Contains information about report job executions
- IS_PROJ: Lookup table for projects
- IS_REP_SQL_STATS: Statistics table containing information about SQL statements

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp when the information was recorded by Intelligence Server into the _STATS table.
IS_PROJ_ID	Integer ID of the project logged into.
IS_PROJ_NAME	Project name.
IS_REP_JOB_SES_ID	GUID of the current session object.
IS_REP_JOB_ID	Integer ID of the report job execution.
IS_PASS_SEQ_NBR	Integer ID of the sequence of the pass.
IS_REP_SQL_SEQ	If a SQL statement is very long, it is broken into multiple rows. This column represents the Sequence of a SQL Statement. For example, if a SQL is very long and broken into two parts, this table would contain two rows for that SQL with the value of this column being '1' and '2'.
IS_REP_EXEC_ST_TS	Timestamp of the execution start.

Column Name	Column Description
IS_REP_EXEC_FN_TS	Timestamp of the execution finish.
IS_REP_EXEC_TM_MS	Execution duration in milliseconds.
IS_REP_SQL_STATEM	SQL statement.
IS_REP_SQL_LENGTH	Length of SQL statement.
IS_REP_NBR_TABLES	Number of tables accessed by SQL statement.
IS_PASS_TYPE_ID	Integer ID of the type of SQL pass.
IS_REP_DB_ERR_MSG	Error returned from the database; NULL if no error.
DAY_ID	Day the job was requested for execution.
HOUR_ID	Hour the job was requested for execution.
MINUTE_ID	Minute the job was requested for execution.
IS_REPOSITORY_ID	Integer ID of the metadata repository.

IS_REP_STEP_FACT

Contains information about the processing steps through which the report execution passes. Created as a view based on columns in the source tables listed below.

Source tables

- IS_REP_STEP_STATS: Statistics table containing information about report job processing steps
- IS_REPORT_STATS: Statistics table containing information about report job executions
- IS_SESSION: Lookup table for session objects
- IS_PROJ: Lookup table for projects

List of table columns

Column Name	Column Description
EM_RECORD_TS	Timestamp when the information was recorded by Intelligence Server into the _STATS table.
IS_PROJ_ID	Integer ID of the project logged into.
IS_PROJ_NAME	Project name.

Column Name	Column Description
IS_REP_JOB_SES_ID	GUID of the current session object.
IS_REP_JOB_ID	Integer ID of the report job execution.
IS_REP_STEP_SEQ_ID	Integer ID of the sequence of the step.
IS_REP_STEP_TYP_ID	Integer ID of the type of step.
IS_REP_EXEC_ST_TS	Timestamp of the execution start.
IS_REP_EXEC_FN_TS	Timestamp of the execution finish.
IS_REP_QU_TM_MS	Queue duration in milliseconds.
IS_REP_CPU_TM_MS	CPU duration in milliseconds.
IS_REP_EXEC_TM_MS	Execution duration in milliseconds.
DAY_ID	Day the job was requested for execution.
HOUR_ID	Hour the job was requested for execution.
MINUTE_ID	Minute the job was requested for execution.
IS_REPOSITORY_ID	Integer ID of the metadata repository.

IS_SESSION_FACT

Enables session concurrency analysis. Keeps data on each session for each hour of connectivity.

Related lookup tables

- IS_SESSION: Lookup table for session objects
- DT_DAY: Lookup table for dates
- TM_HOUR: Lookup table for hours

List of table columns

Column Name	Column Description
IS_SESSION_ID	GUID of the session object.
IS_SERVER_ID	Integer ID of the server where the session was created.
EM_USER_ID	Integer ID of the user who created the session.
IS_CONNECT_TS	Timestamp of the beginning of the session (login).

Column Name	Column Description
IS_DISCONNECT_TS	Timestamp of the end of the session (logout). NULL if the session is still open at the time of Enterprise Manager data load.
IS_CONNEC_M_ID	Integer representation of the day and hour when the connection began. Format: YYYYMMDDHH (24 hours).
IS_DISCON_M_ID	Integer representation of the day and hour when the connection ended. Format: YYYYMMDDHH (24 hours).

Column Name	Column Description
EM_CONNECT_SOURCE	Connection source through which the session was established: • 0: Unknown • 1: MicroStrategy Developer • 2: MicroStrategy Intelligence Server Administrator • 3: MicroStrategy Web Administrator • 4: MicroStrategy Intelligence Server • 5: MicroStrategy Project Upgrade • 6: MicroStrategy Web • 7: MicroStrategy Scheduler • 8: Custom application • 9: MicroStrategy Narrowcast Server • 10: MicroStrategy Object Manager • 11: ODBO Provider • 12: ODBO Cube Designer • 13: MicroStrategy Command Manager • 14: MicroStrategy Enterprise Manager • 15: MicroStrategy Command Line Interface • 16: MicroStrategy Project Builder • 17: MicroStrategy Configuration Wizard • 18: MicroStrategy MD Scan • 19: MicroStrategy Cache Utility • 20: MicroStrategy Fire Event • 21: MicroStrategy Java Admin Clients • 22: MicroStrategy Web Services • 23: MicroStrategy Office • 24: MicroStrategy Tools • 25: MicroStrategy Portal Server • 26: MicroStrategy Integrity Manager • 27: Metadata Update • 28: COM Browser • 29: MicroStrategy Mobile • 30: Repository Translation Wizard • 31: MicroStrategy Health Center

Column Name	Column Description
	• 32: MicroStrategy Cube Advisor
DAY_ID	Integer ID of the day. Format: YYYYMMDD.
HOUR_ID	Integer ID of the hour. Format HH (24 hours).
MINUTE_ID	Minute the job was executed.

IS_SESSION_MONITOR

For MicroStrategy use. A view table that provides an overview of recent session activity.

Lookup tables

Table Name	Function
CT_EXEC_TYPE	Lookup table for mobile client execution type. • 0: Unknown • 1: User • 2: Pre-cached • 3: Application recovery • 4: Subscription cache pre-loading • 5: Transaction subsequent action • 6: Report queue • 7: Report queue recall • 8: Back button

Table Name	Function
CT_MANIP_TYPE	Lookup table for mobile client manipulation type: • 0: Unknown • 1: Selector • 2: Panel Selector • 3: Action Selector • 4: Change Layout • 5: Change View • 6: Sort • 7: Page By • 8: Information Window • 9: Annotations • 10: E-mail Screenshots • 11: Widget: Video-Play • 12: Widget: Video-Pause • 20: Widget: Multiple-Download • 21: Widget: Multiple-Open
DT_DAY	Lookup table for Days in the Date hierarchy.
DT_MONTH	Lookup table for Months in the Date hierarchy.
DT_MONTH_OF_YR	Lookup table for Months of the Year in the Date hierarchy.
DT_QUARTER	Lookup table for Quarters in the Date hierarchy.
DT_QUARTER_OF_YR	Lookup table for the Quarters of the Year in the Date hierarchy.
DT_WEEKDAY	Lookup table for the Days of the Week in the Date hierarchy.
DT_WEEKOFYEAR	Lookup table for Weeks of the Year in the Date hierarchy.
DT_YEAR	Lookup table for Years in the Date hierarchy.
EM_APP_SRV_MACHINE	Lookup table for Intelligence Server machines used in statistics.
EM_CLIENT_MACHINE	Lookup table for Client Machines used in the statistics.
EM_CONNECT_SOURCE	Lookup table for the connection source of a session on Intelligence Server.
EM_DB_USER	Lookup table for the database users used in the statistics.
EM_EXISTS_IND	Lookup table for the existence status of objects.
EM_HIDDEN_IND	Lookup table for the hidden status of objects.
EM_JOB_STATUS	Lookup table for the job status of job executions on Intelligence Server:

Table Name	Function
	• 0: Ready • 1: Executing • 2: Waiting • 3: Completed • 4: Error • 5: Cancelled • 6: Stopped • 7: Waiting for governor • 8: Waiting for prompt • 9: Waiting for project • 10: Waiting for cache • 11: Waiting for children • 12: Waiting for fetching results
EM_MONITORED_ PROJECTS	Provides information about the projects being monitored and when the first and last data loads occurred.
EM_OWNER	Provides descriptive information about the owners being tracked. This table is a view based on columns from the EM_USER table.
EM_USER	Provides descriptive information about the users being tracked.
EM_USR_GP	Provides descriptive information about the user groups being tracked.
EM_WEB_SRV_MACHINE	Lookup table for the Web Server Machines used in the statistics.
IS_AD_HOC_IND	Lookup table for the Ad Hoc indicator.
IS_ATT	Provides descriptive information about the attributes being tracked.
IS_ATT_FORM	Provides descriptive information about the attribute forms being tracked.
IS_CACHE_CREATION_IND	Lookup table for the Cache Creation indicator.
IS_CACHE_HIT_TYPE	Lookup table for the Cache Hit indicator: • 0: No cache hit • 1: Server cache • 2: Device cache • 6: Application cache
IS_CANCELLED_IND	Lookup table for the Canceled indicator.
IS_CHILD_JOB_IND	Lookup table for the Child Job indicator.

Table Name	Function
IS_COL	Provides descriptive information about the columns being tracked.
IS_CONFIG_PARAM	Lookup table for Intelligence Server and project-level configuration settings.
IS_CONS	Provides descriptive information about the consolidations being tracked.
IS_CONTACT_TYPE	Lookup table for the type of contact delivered to through Distribution Services.
IS_CUBE_ACTION_TYPE	Lookup table for the manipulations that can be performed on an Intelligent Cube.
IS_CUBE_HIT_IND	Lookup table for the Cube Hit indicator.
IS_CUBE_VIEW	Provides descriptive information about the Intelligent Cubes being tracked. This table is a view based on columns from the IS_REP table.
IS_CUST_GP	Provides descriptive information about the custom groups being tracked.
IS_DATA_TYPE	Lookup table for the job type: • 3: Report • 55: Document
IS_DATAMART_IND	Lookup table for the Data Mart indicator.
IS_DB_ERROR_IND	Lookup table for the Database Error indicator.
IS_DB_INST	Provides descriptive information about the database instances in the monitored Intelligence Servers.
IS_DB_TAB	Lookup table for the database tables being monitored.
IS_DELIVERY_STATUS_IND	Lookup table for the Delivery Status indicator.
IS_DELIVERY_TYPE	Lookup table for the Distribution Services delivery type.
IS_DEVICE	Provides descriptive information about the devices being tracked.
IS_DOC	Provides descriptive information about the document objects being tracked.
IS_DOC_STEP_TYPE	Lookup table for the step types in document execution. For a list and explanation of values, see *Lookup tables, page 445*.
IS_DOCTYPE_IND	Indicator lookup table for document or dashboard type. Types include: • -1: Unknown • 0: HTML document • 1: Report Services document • 2: Visual Insight dashboard

Table Name	Function
IS_DRILL_IND	Lookup table for the Drill indicator.
IS_ELEM_LOAD_IND	Lookup table for the Element Load indicator.
IS_ERROR_IND	Lookup table for the Error indicator.
IS_EVENT	Provides descriptive information about the events being tracked.
IS_EXPORT_IND	Lookup table for the Export indicator.
IS_FACT	Provides descriptive information about the facts being tracked.
IS_FILT	Provide descriptive information about the filters being tracked.
IS_HIER	Provides descriptive information about the hierarchies being tracked.
IS_HIER_DRILL_IND	Lookup table for the Drillable Hierarchy indicator.
IS_INBOX_ACTION	Provides a list of the different manipulations that can be performed on a History List message.
IS_JOB_PRIORITY_TYPE	Lookup table for the Job Priority type.
IS_MET	Provides descriptive information about the metrics being tracked.
IS_OLAP_CUBE	Provides descriptive information about the Intelligent Cubes being tracked.
IS_PRIORITY_MAP	Indicator lookup table for priority maps.
IS_PROJ	Provides descriptive information about the projects being tracked.
IS_PROMPT	Provides descriptive information about the prompts being tracked.
IS_PROMPT_IND	Lookup table for the Prompt indicator.
IS_REP	Provides descriptive information about the reports being tracked.
IS_REP_SQL_PASS_TYPE	Lookup table for the SQL pass types of report execution.
IS_REP_STEP_TYPE	Lookup table for the step types of report execution. For a list and explanation of values, see *Lookup tables, page 445*.
IS_REPCTYPE_IND	Lookup table for the Report Cube Type indicator: • 0: Reserved • 1: Base Report • 2: Working Set Report • 3: Private Base Report • 5: Report Services Base Report • 6: CSQL Pre-Execution Report • 7: OLAP Cube Report • 8: OLAP View Report

Table Name	Function
	• 9: Incremental Refresh Report
IS_REPTYPE_IND	Indicator lookup table for report type. Report types include: • -1: Unknown: The server is unable to retrieve the report type. • 0: Reserved: Ad hoc reports. May include other reports that are not persisted in the metadata at the point of execution. • 1: Relational: All regular project reports. • 2: MDX: Reports built from SAP BW, Essbase, Analysis Services, and other cube sources. • 3: Custom SQL Freeform: MicroStrategy Freeform SQL reports, in which the SQL is entered directly into the interface. • 4: Custom SQL Wizard: MicroStrategy Query Builder reports. • 5: Flat File: Reserved for MicroStrategy use.
IS_SCHED	Provides descriptive information about the schedules being tracked.
IS_SCHEDULE_IND	Lookup table for the Schedule indicator.
IS_SEC_FILT	Provides descriptive information about the security filters being tracked.
IS_SEC_FILT_IND	Lookup table for the Security Filter indicator.
IS_SERVER	Provides descriptive information about the server definitions being tracked.
IS_SESSION	Lookup table for the session statistics logged by Intelligence Servers. Primary key: DAY_ID, IS_SESSION_ID
IS_SQL_CLAUSE_TYPE	Lookup table for SQL clause types; used to determine which SQL clause (SELECT, WHERE, GROUP BY, and so on) a particular column was used in during a report execution. SQL Clause Type attributes: • 1: Select: Column was used in the SELECT clause but was not aggregated, nor does it appear in a GROUP BY clause. For example, a11.Report column in “Select a11.Report from LU_REPORT a11”. • 2: Select Group By: Column was used in the GROUP BY clause. For example, a11.Report Column in “select a11.Report, sum(a11.Profit) from LU_REPORT group by a11.Report”. • 4: Select Aggregate: Column was used for aggregation. For example, a11.Report column in “select count(a11.Report) from LU_REPORT”. • 8: From: Column was used in a FROM clause • 16: Where: Column was used in a WHERE clause • 17: Order By: Column was used in an ORDER BY clause.
IS_SQL_EXEC_IND	Lookup table for the SQL Execution indicator.

Table Name	Function
IS_TABLE	Provides descriptive information about the logical tables being monitored.
IS_TEMP	Provides descriptive information about the templates being monitored.
IS_TRANS	Provides descriptive information about the transformations being monitored.
IS_TRANS_MAP	Lookup table for the transformation mapping types.
IS_TRANSMIT	Provides descriptive information about the information transmitters being monitored.
TM_HOUR	Lookup table for Hour in the Time hierarchy.
TM_MINUTE	Lookup table for Minute in the Time hierarchy.

Transformation tables

Table Name	Function
DT_MONTH_YTD	Transformation table to calculate the Year to Date values for Month.
DT_QUARTER_YTD	Transformation table to calculate the Year to Date values for Quarter.
TM_HOUR_DTH	Transformation table to calculate the Hour to Day values for Hour.

Report and document steps

This IS_REP_STEP_TYPE table lists the Intelligence Server tasks involved in executing a report or a document. These are the possible values for the IS_REP_STEP_TYP_ID column in the IS_REP_STEP_STATS table and the IS_DOC_STEP_TYP_ID column in the IS_DOC_STEP_STATS table.

Not all steps are applicable to all types of reports. For example, if you are not using Intelligent Cubes, those steps are skipped.

Task name	Task description
0: Unknown	Reserved for MicroStrategy use.
1: MD Object Request	The Object Server component in Intelligence Server requests the objects necessary for the report.
2: Close Job	Intelligence Server closes the report execution job.
3: SQL Generation	The SQL Engine generates the SQL to be executed against the data warehouse.

Task name	Task description
4: SQL Execution	The Query Engine submits the generated SQL to the data warehouse, and receives the result.
5: Analytical Engine	The Analytical Engine applies additional processing steps to the data retrieved from the warehouse.
6: Resolution Server	The Resolution Server uses the report definition to retrieve objects from the Object Server.
7: Report Net Server	The Report Net Server processes report requests and sends them to the Report Server.
8: Element Request	The Resolution Server works with the Object Server and Element Server to resolve prompts for report requests.
9: Get Report Instance	Intelligence Server receives the report instance from the Report Server.
10: Error Message Send	If an error occurs, Intelligence Server sends a message to the user, and logs the error.
11: Output Message Send	When the report finishes executing, the output data is sent to the client.
12: Find Report Cache	The Report Server searches the cache for a previously run report.
13: Document Execution	Intelligence Server executes the datasets needed for the document, and creates the document structure.
14: Document Send	Once a document is executed, Intelligence Server sends the output to the client (such as MicroStrategy Developer or Web).
15: Update Report Cache	Once a report is executed, the Report Server writes the data to the report cache.
16: Request Execute	The client (such as MicroStrategy Developer or Web) requests the execution of a report or document.
17: Data Mart Execute	The Query Engine executes the SQL to create the data mart table.
18: Document Data Preparation	Intelligence Server prepares the document data, performing tasks such as dataset joins, where applicable.
19: Document Formatting	Intelligence Server combines the data for the document with the structure, and formats the output.
20: Document Manipulation	Intelligence Server applies the user's manipulations to a document.
21: Apply View Context	Intelligence Server executes a view report against an Intelligent Cube.
22: Export Engine	The Export Engine formats a report or document for export as a PDF, Excel workbook, or XML.
23: Find Intelligent Cube	The SQL Engine matches a view report, or a report that uses dynamic sourcing, with the corresponding Intelligent Cube.

Task name	Task description
24: Update Intelligent Cube	The Query Engine runs the SQL required to refresh the data in the Intelligent Cube.
25: Post-processing Task	Reserved for MicroStrategy use.
26: Delivery	Distribution Services delivers the report to email, files, printers, or mobile.
27: Persist Result	Intelligence Server checks if the conditions for alert-based subscriptions are met. If so, the subscribed report is executed and delivered. If the condition is not met, the job is cancelled.
28: Document Dataset Execution	The document is waiting for its dataset report jobs to finish executing.

Relationship tables

Table Name	Function
IS_ATT_ATT_ASSOCS	Relationship table containing attributes and associated attributes, for affinity analysis.
IS_ATT_ATT_FORM	Relationship table between Attribute and Attribute Form.
IS_ATT_HIER	Relationship table between Attribute and Hierarchy.
IS_CNDT_REP	Relationship table between Intelligent Cube and reports that could hit an Intelligent Cube.
IS_COL_TABLE	Relationship table between Column and Table.
IS_MET_ATT_ASSOCS	Relationship table containing metrics and associated attributes, for affinity analysis.
IS_MET_TEMP	Relationship table between Metric and Template.
IS_REP_ATT	Relationship table for reports and component attributes.
IS_REP_CONS	Relationship table between Consolidation and Report.
IS_REP_DOC	Relationship table between Report and Document.
IS_REP_FILT	Relationship table between Filter and Report.
IS_REP_MET	Relationship table for reports and component metrics.
IS_REP_PROMPT	Relationship table between Prompt and Report.
IS_REP_TEMPLATE	Relationship table between Template and Report.
IS_SCHED_REL_DOC	Relationship table for schedules and associated documents.
IS_SCHED_RELATE	Relationship table for schedules and associated reports.

Table Name	Function
IS_TABLE_FACT	Relationship table between Table and Fact.
IS_TEMP_ATT	Relationship table between Template and Attribute.
IS_USER_PROJ_SF	Relationship table for users and associated security filters.
IS_USR_GP_USER	Relationship table between User and User Group.
IS_USR_GP_USR_GP	Relationship table between User Group and User Group (Parent).

Enterprise Manager metadata tables

The following is a description of Enterprise Manager metadata tables.

Table Name	Function
EM_COMP	Defines all MicroStrategy components being monitored. The abbreviation specifies the prefix used on tables relevant to the component. When a new component is added to the MicroStrategy product line, it can be entered in this table for monitoring. **Examples**: Intelligence Server, Narrowcast Server
EM_IS_LAST_UPDATE	Provides the Data Loading process with a working window that identifies the period during which data should be moved into production area tables.
EM_ITEM	Defines all items in each component of the MicroStrategy product line being monitored. When a new item is added to a component, it can be entered in this table for monitoring, without any change to the migration code. This table also specifies the item's object type according to server and the abbreviation used in the lookup table name. **Examples**: Report, Server Definition, User
EM_ITEM_PROPS	Identifies properties being tracked on a given item for a given component. The SQL for the item is a concatenation of the general SQL in the EM_ITEM_SQL table and the column names and values stored in this table. **Examples**: Attribute Number of Parents, Hierarchy Drill Enabled Primary key: EM_COMP_ID, EM_ITEM_ID, EM_PROP_ID
EM_LOG	Stores logging information for Enterprise Manager data loads. The logging option is enabled from the Enterprise Manager console, **Tools** menu, **Options** selection. Data warehouse purges are not logged in this table.
EM_PROPS	Shows properties of the Enterprise Manager application (for example: which projects and servers are being tracked).
EM_RELATE_ITEM	Contains a list of many-to-many relationship tables and the MicroStrategy items they relate.

Table Name	Function
EM_SQL	Provides the SQL necessary to insert, update, and delete a row from the lookup item table once the necessary information from the component API is available. If the SQL must be changed, make the change in this table (no changes in the code are necessary). This table also provides the SQL used to transform the logged statistics into the lookup tables. **Example**: SQL statements to insert an attribute into the lookup attribute table in SQL Server

Relationship tables

Table Name	Function
IS_ATT_ATT_ASSOCS	Relationship table containing attributes and associated attributes, for affinity analysis.
IS_ATT_ATT_FORM	Relationship table between Attribute and Attribute Form.
IS_ATT_HIER	Relationship table between Attribute and Hierarchy.
IS_CNDT_REP	Relationship table between Intelligent Cube and reports that could hit an Intelligent Cube.
IS_COL_TABLE	Relationship table between Column and Table.
IS_MET_ATT_ASSOCS	Relationship table containing metrics and associated attributes, for affinity analysis.
IS_MET_TEMP	Relationship table between Metric and Template.
IS_REP_ATT	Relationship table for reports and component attributes.
IS_REP_CONS	Relationship table between Consolidation and Report.
IS_REP_DOC	Relationship table between Report and Document.
IS_REP_FILT	Relationship table between Filter and Report.
IS_REP_MET	Relationship table for reports and component metrics.
IS_REP_PROMPT	Relationship table between Prompt and Report.
IS_REP_TEMPLATE	Relationship table between Template and Report.
IS_SCHED_REL_DOC	Relationship table for schedules and associated documents.
IS_SCHED_RELATE	Relationship table for schedules and associated reports.
IS_TABLE_FACT	Relationship table between Table and Fact.
IS_TEMP_ATT	Relationship table between Template and Attribute.
IS_USER_PROJ_SF	Relationship table for users/groups and associated security filters.
IS_USR_GP_USER	Relationship table between User and User Group.
IS_USR_GP_USR_GP	Relationship table between User Group and User Group (Parent).

Enterprise Manager metadata tables

The following is a description of Enterprise Manager metadata tables.

Table Name	Function
EM_COMP	Defines all MicroStrategy components being monitored. The abbreviation specifies the prefix used on tables relevant to the component. When a new component is added to the MicroStrategy product line, it can be entered in this table for monitoring. **Examples**: Intelligence Server, Narrowcast Server
EM_IS_LAST_UPDATE	Provides the Data Loading process with a working window that identifies the period during which data should be moved into production area tables.
EM_ITEM	Defines all items in each component of the MicroStrategy product line being monitored. When a new item is added to a component, it can be entered in this table for monitoring, without any change to the migration code. This table also specifies the item's object type according to server and the abbreviation used in the lookup table name. **Examples**: Report, Server Definition, User
EM_ITEM_PROPS	Identifies properties being tracked on a given item for a given component. The SQL for the item is a concatenation of the general SQL in the EM_ITEM_SQL table and the column names and values stored in this table. **Examples**: Attribute Number of Parents, Hierarchy Drill Enabled Primary key: EM_COMP_ID, EM_ITEM_ID, EM_PROP_ID
EM_LOG	Stores logging information for Enterprise Manager data loads. The logging option is enabled from the Enterprise Manager console, **Tools** menu, **Options** selection. Data warehouse purges are not logged in this table.
EM_PROPS	Shows properties of the Enterprise Manager application (for example: which projects and servers are being tracked).
EM_RELATE_ITEM	Contains a list of many-to-many relationship tables and the MicroStrategy items they relate.
EM_SQL	Provides the SQL necessary to insert, update, and delete a row from the lookup item table once the necessary information from the component API is available. If the SQL must be changed, make the change in this table (no changes in the code are necessary). This table also provides the SQL used to transform the logged statistics into the lookup tables. **Example**: SQL statements to insert an attribute into the lookup attribute table in SQL Server

Enterprise Manager attributes and metrics

The following sections list the contents of the Enterprise Manager attributes folders. These include attributes and shortcuts to metrics that are useful in creating reports in the Enterprise Manager project. The items in the folders are grouped by the type of reporting you can do with them.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.

Attribute name	Function
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.

Attribute name	Function
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.

Attribute or metric name	Function
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.

Attribute or metric name	Function
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_ UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.

Attribute or metric name	Function
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.

Attribute name	Function
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.

Attribute or metric name	Function
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.

Attribute or metric name	Function
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.

Attribute or metric name	Function
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution	Metric of the average duration, in seconds, of all report job

Attribute or metric name	Function
Duration per Job (secs) (IS_REP_FACT)	executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.

Attribute or metric name	Function
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.

Attribute or metric name	Function
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.

Attribute or metric name	Function
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.

Attribute or metric name	Function
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute name	Function
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.

Attribute name	Function
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise

Attribute name	Function
	Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.

Attribute or metric name	Function
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job

Attribute or metric name	Function
	executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.

Attribute or metric name	Function
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.

Attribute name	Function
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.

Attribute name	Function
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.

Attribute name	Function
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.

Attribute name	Function
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration	Metric of the average amount of time subscriptions take to

Attribute or metric name	Function
(hh:mm:ss)	execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.

Attribute or metric name	Function
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.

Attribute or metric name	Function
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.

Attribute name	Function
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_ UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.

Attribute or metric name	Function
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.

Attribute name	Function
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting	Metric of how many users executed a report or document that used

Attribute or metric name	Function
Intelligent Cubes	an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.

Attribute or metric name	Function
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.

Attribute or metric name	Function
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.

Attribute or metric name	Function
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.

Attribute or metric name	Function
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute or metric name	Function
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.

Attribute or metric name	Function
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.

Attribute name	Function
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.

Attribute name	Function
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.

Attribute name	Function
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.

Attribute or metric name	Function
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.

Attribute or metric name	Function
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.

Attribute or metric name	Function
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.

Attribute or metric name	Function
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.

Attribute name	Function
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.

Attribute name	Function
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.

Attribute name	Function
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.

Attribute name	Function
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration	Metric of the average amount of time subscriptions take to

Attribute or metric name	Function
(hh:mm:ss)	execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.

Attribute or metric name	Function
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.

Attribute or metric name	Function
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.

Attribute name	Function
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.

Attribute or metric name	Function
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.

Attribute name	Function
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting	Metric of how many users executed a report or document that used

Attribute or metric name	Function
Intelligent Cubes	an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.

Attribute or metric name	Function
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.

Attribute or metric name	Function
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.

Attribute or metric name	Function
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.

Attribute or metric name	Function
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute or metric name	Function
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.

Attribute or metric name	Function
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.

Attribute name	Function
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.

Attribute name	Function
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.

Attribute name	Function
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.

Attribute or metric name	Function
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.

Attribute or metric name	Function
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.

Attribute or metric name	Function
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.

Attribute or metric name	Function
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.

Attribute name	Function
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.

Attribute name	Function
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.

Attribute name	Function
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.

Attribute name	Function
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration	Metric of the average amount of time subscriptions take to

Attribute or metric name	Function
(hh:mm:ss)	execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.

Attribute or metric name	Function
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.

Attribute or metric name	Function
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.

Attribute name	Function
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.

Attribute or metric name	Function
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.

Attribute name	Function
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting	Metric of how many users executed a report or document that used

Attribute or metric name	Function
Intelligent Cubes	an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.

Attribute or metric name	Function
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.

Attribute or metric name	Function
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.

Attribute or metric name	Function
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.

Attribute or metric name	Function
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute or metric name	Function
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.

Attribute or metric name	Function
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.

Attribute name	Function
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.

Attribute name	Function
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.

Attribute name	Function
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.

Attribute or metric name	Function
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.

Attribute or metric name	Function
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.

Attribute or metric name	Function
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.

Attribute or metric name	Function
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.

Attribute name	Function
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.

Attribute name	Function
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.

Attribute name	Function
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.

Attribute name	Function
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration	Metric of the average amount of time subscriptions take to

Attribute or metric name	Function
(hh:mm:ss)	execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.

Attribute or metric name	Function
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.

Attribute or metric name	Function
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.

Attribute name	Function
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.

Attribute or metric name	Function
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.

Attribute name	Function
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting	Metric of how many users executed a report or document that used

Attribute or metric name	Function
Intelligent Cubes	an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.

Attribute or metric name	Function
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.

Attribute or metric name	Function
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.

Attribute or metric name	Function
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.

Attribute or metric name	Function
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute or metric name	Function
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.

Attribute or metric name	Function
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.

Attribute name	Function
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.

Attribute name	Function
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.

Attribute name	Function
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.

Attribute or metric name	Function
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.

Attribute or metric name	Function
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.

Attribute or metric name	Function
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.

Attribute or metric name	Function
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.

Attribute name	Function
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.

Attribute name	Function
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.

Attribute name	Function
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.

Attribute name	Function
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration	Metric of the average amount of time subscriptions take to

Attribute or metric name	Function
(hh:mm:ss)	execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.

Attribute or metric name	Function
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.

Attribute or metric name	Function
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.

Attribute name	Function
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.

Attribute or metric name	Function
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.

Attribute name	Function
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting	Metric of how many users executed a report or document that used

Attribute or metric name	Function
Intelligent Cubes	an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.

Attribute or metric name	Function
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.

Attribute or metric name	Function
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.

Attribute or metric name	Function
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.

Attribute or metric name	Function
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute or metric name	Function
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.

Attribute or metric name	Function
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.

Attribute name	Function
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.

Attribute name	Function
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.

Attribute name	Function
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.

Attribute or metric name	Function
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.

Attribute or metric name	Function
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.

Attribute or metric name	Function
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.

Attribute or metric name	Function
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.

Attribute name	Function
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.

Attribute name	Function
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.

Attribute name	Function
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.

Attribute name	Function
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration	Metric of the average amount of time subscriptions take to

Attribute or metric name	Function
(hh:mm:ss)	execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.

Attribute or metric name	Function
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.

Attribute or metric name	Function
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.

Attribute name	Function
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.

Attribute or metric name	Function
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.

Attribute name	Function
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting	Metric of how many users executed a report or document that used

Attribute or metric name	Function
Intelligent Cubes	an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.

Attribute or metric name	Function
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.

Attribute or metric name	Function
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.

Attribute or metric name	Function
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.

Attribute or metric name	Function
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute or metric name	Function
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.

Attribute or metric name	Function
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.

Attribute name	Function
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.

Attribute name	Function
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.

Attribute name	Function
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.

Attribute or metric name	Function
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.

Attribute or metric name	Function
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.

Attribute or metric name	Function
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.

Attribute or metric name	Function
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.

Attribute name	Function
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.

Attribute name	Function
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.

Attribute name	Function
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.

Attribute name	Function
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration	Metric of the average amount of time subscriptions take to

Attribute or metric name	Function
(hh:mm:ss)	execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.

Attribute or metric name	Function
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.

Attribute or metric name	Function
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.

Attribute name	Function
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.

Attribute or metric name	Function
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.

Attribute name	Function
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting	Metric of how many users executed a report or document that used

Attribute or metric name	Function
Intelligent Cubes	an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.

Attribute or metric name	Function
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.

Attribute or metric name	Function
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.

Attribute or metric name	Function
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.

Attribute or metric name	Function
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute or metric name	Function
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.

Attribute or metric name	Function
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.

Attribute name	Function
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.

Attribute name	Function
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.

Attribute name	Function
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.

Attribute or metric name	Function
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.

Attribute or metric name	Function
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.

Attribute or metric name	Function
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.

Attribute or metric name	Function
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.

Attribute name	Function
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.

Attribute name	Function
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.

Attribute name	Function
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.

Attribute name	Function
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration	Metric of the average amount of time subscriptions take to

Attribute or metric name	Function
(hh:mm:ss)	execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.

Attribute or metric name	Function
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.

Attribute or metric name	Function
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.

Attribute name	Function
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.

Attribute or metric name	Function
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.

Attribute name	Function
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting	Metric of how many users executed a report or document that used

Attribute or metric name	Function
Intelligent Cubes	an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.

Attribute or metric name	Function
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.

Attribute or metric name	Function
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.

Attribute or metric name	Function
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.

Attribute or metric name	Function
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute or metric name	Function
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.

Attribute or metric name	Function
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.

Attribute name	Function
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.

Attribute name	Function
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.

Attribute name	Function
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.

Attribute or metric name	Function
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.

Attribute or metric name	Function
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.

Attribute or metric name	Function
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.

Attribute or metric name	Function
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.

Attribute name	Function
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.

Attribute name	Function
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.

Attribute name	Function
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.

Attribute name	Function
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration	Metric of the average amount of time subscriptions take to

Attribute or metric name	Function
(hh:mm:ss)	execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.

Attribute or metric name	Function
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.

Attribute or metric name	Function
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.

Attribute or metric name	Function
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.

Attribute name	Function
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_ UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.

Attribute or metric name	Function
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.

Attribute name	Function
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting	Metric of how many users executed a report or document that used

Attribute or metric name	Function
Intelligent Cubes	an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.

Attribute or metric name	Function
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.

Attribute or metric name	Function
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.

Attribute or metric name	Function
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.

Attribute or metric name	Function
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.

Attribute or metric name	Function
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.

Attribute or metric name	Function
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

All Indicators and Flags attributes

Attribute name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been cancelled.
Child Job Indicator	Indicates whether a job was a document dataset or a stand-alone report.
Configuration Object Exists Status	Indicates whether a configuration object exists.
Configuration Parameter Value Type	Lists all configuration parameter types.
Connection Source	Lists all connection sources to Intelligence Server.
Contact Type	Lists the executed contact types.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
DB Error Indicator	Indicates whether an execution encountered a database error.
Delivery Status Indicator	Indicates whether a delivery was successful.
Delivery Type	Lists the type of delivery.

Attribute name	Function
Document Job Step Type	Lists all possible steps of document job execution.
Document Type	Indicates the type of a document or dashboard, such as HTML document, Report Services document, or Visual Insight dashboard.
Drill from Object	Lists the object from which a user drilled when a new report was run because of a drilling action.
Drill Indicator	Indicates whether an execution is a result of a drill.
Drill to Object	Lists the object to which a user drilled when a new report was run because of a drilling action.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Execution Type Indicator	Indicates how the content was requested, such as User Execution, Pre-Cached, Application Recovery, and so on.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Hierarchy Drilling	Indicates whether a hierarchy is used as a drill hierarchy.
Inbox Action Type	List the types of manipulations that can be performed on a History List message.
Intelligent Cube Action Type	Lists actions performed on or against intelligent cubes.
Intelligent Cube Type	Lists all intelligent cube types.
Job ErrorCode	Lists all the possible errors that can be returned during job executions.
Job Priority Map	Lists the priorities of job executions.
Job Priority Number	Enumerates the upper limit of the priority ranges for high, medium, and low priority jobs. Default values are 332, 666, and 999.
Object Creation Date	Indicates the date on which an object was created.
Object Creation Week of year	Indicates the week of the year in which an object was created.
Object Exists Status	Indicates whether an object exists.
Object Hidden Status	Indicates whether an object is hidden.
Object Modification Date	Indicates the date on which an object was last modified.
Object Modification Week of year	Indicates the week of the year in which an object was last modified.
Prompt Answer Required	Indicates whether a prompt answer was required for the job execution.
Prompt Indicator	Indicates whether a job execution was prompted.
Report Job SQL Pass Type	Lists the types of SQL passes that the Intelligence Server generates.

Attribute name	Function
Report Job Status	Lists the statuses of report executions.
Report Job Step Type	Lists all possible steps of report job execution.
Report Type	Indicates the type of a report, such as XDA, relational, and so on.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule Indicator	Indicates whether a job execution was scheduled.
Security Filter Indicator	Indicates whether a security filter was used in the job execution.
SQL Clause Type	Lists the various SQL clause types used by the SQL Engine.
SQL Execution Indicator	Indicates whether SQL was executed in the job execution.

Application Objects attributes

Attribute name	Function
Consolidation	Lists all consolidations in projects that are set up to be monitored by Enterprise Manager.
Custom Group	Lists all custom groups in projects that are set up to be monitored by Enterprise Manager.
Document	Lists all documents in projects that are set up to be monitored by Enterprise Manager.
Filter	Lists all filters in projects that are set up to be monitored by Enterprise Manager.
Intelligent Cube	Lists all intelligent cubes in projects that are set up to be monitored by Enterprise Manager.
Metric	Lists all metrics in projects that are set up to be monitored by Enterprise Manager.
Prompt	Lists all prompts in projects that are set up to be monitored by Enterprise Manager.
Report	Lists all reports in projects that are set up to be monitored by Enterprise Manager.
Security Filter	Lists all security filters in projects that are set up to be monitored by Enterprise Manager.
Template	Lists all templates in projects that are set up to be monitored by Enterprise Manager.

Configuration Objects attributes

Attribute name	Function
Address	Lists all addresses to which deliveries have been sent.
Configuration Object Owner	Lists the owners of configuration objects.
Configuration Parameter	Lists all configuration parameters.
Contact	Lists all contacts to whom deliveries have been sent.
DB Connection	Lists all database connections.
DB Instance	Lists all database instances.
Device	Lists all devices to which deliveries have been sent.
Event	Lists all events being tracked.
Folder	Lists all folders within projects.
Intelligence Server Definition	Lists all Intelligence Server definitions.
Metadata	Lists all monitored metadata.
Owner	Lists the owners of all objects.
Project	Lists all projects.
Schedule	Lists all schedules.
Subscription	Lists all executed transmissions.
Transmitter	Lists all transmitters.
User	Lists all users being tracked.
User Group	Lists all user groups.
User Group (Parent)	Lists all user groups that are parents of other user groups.

Date and Time attributes

Attribute name	Function
Calendar Week	Lists every calendar week, beginning with 2000-01-01, as an integer.
Day	Lists all days, beginning in 1990.
Hour	Lists the hours in a day. For example, 09 AM - 10 AM, 10 AM - 11 AM, and so on.
Minute	Lists all the minutes in an hour. For example, if the hour specified is 10 AM - 11 AM, lists minutes as 10.30 AM - 10.31 AM, 10.32 AM - 10.33 AM, and so on.

Attribute name	Function
Month	Lists all months, beginning with 2000.
Month of Year	Lists all months in a specified year.
Quarter	Lists all quarters.
Quarter of Year	Lists all quarters of the year.
Week of Year	Lists all weeks in all years, beginning in 2000. Weeks in 2000 are represented as a number ranging from 200001 to 200053, weeks in 2001 are represented as a number ranging from 200101 to 200153, and so on.
Weekday	Lists all days of the week.
Year	Lists all years.

Delivery Services attributes and metrics

Attribute or metric name	Function
Address	Indicates the address to which a delivery was sent.
Avg number of recipients per subscription	Metric of the average number of recipients in subscriptions.
Avg Subscription Execution Duration (hh:mm:ss)	Metric of the average amount of time subscriptions take to execute.
Avg Subscription Execution Duration (secs)	Metric of the average amount of time, in seconds, subscriptions take to execute.
Contact	Indicates all contacts to whom a delivery was sent.
Contact Type	Indicates the executed contact types.
Day	Indicates the day on which the delivery was sent.
Delivery Status Indicator	Indicates whether the delivery was successful.
Delivery Type	Indicates the type of delivery.
Device	Indicates the type of device to which the delivery was sent.
Document	Indicates the document that was delivered.
Hour	Indicates the hour on which the delivery was sent.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the monitored metadata.
Minute	Indicates the minute on which the delivery was sent.

Attribute or metric name	Function
Number of Distinct Document Subscriptions	Metric of the number of report services document subscriptions.
Number of Distinct Recipients	Metric of the number of recipients that received content from a subscription.
Number of Distinct Report Subscriptions	Metric of the number of report subscriptions.
Number of Distinct Subscriptions	Metric of the number of executed subscriptions. This does not reflect the number of subscriptions in the metadata.
Number of E-mail Subscriptions	Metric of the number of subscriptions that delivered content via e-mail.
Number of Errored Subscriptions	Metric of the number of subscriptions that failed.
Number of Executions	Metric of the number of executions of a subscription.
Number of File Subscriptions	Metric of the number of subscriptions that delivered content via file location.
Number of History List Subscriptions	Metric of the number of subscriptions that delivered content via the history list.
Number of Mobile Subscriptions	Metric of the number of subscriptions that delivered content via mobile.
Number of Print Subscriptions	Metric of the number of subscriptions that delivered content via a printer.
Project	Lists the projects.
Report	Lists the reports in projects.
Report Job	Lists an execution of a report.
Report/Document Indicator	Indicates whether the execution was a report or a document.
Schedule	Indicates the schedule that triggered the delivery.
Subscription	Indicates the subscription that triggered the delivery.
Subscription Execution Duration (hh:mm:ss)	Metric of the sum of all execution times of a subscription.
Subscription Execution Duration (secs)	Metric of the sum of all execution times of a subscription (in seconds).

Document Job attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.

Attribute or metric name	Function
Document	Indicates which document was executed.
Document Job	Indicates an execution of a document.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Number of Jobs (IS_DOC_FACT)	Metric of the number of document jobs that were executed.
DP Number of Jobs with Cache Hit	Metric of the number of document jobs that hit a cache.
DP Number of Jobs with Error	Metric of the number of document jobs that failed.
DP Number of Users who ran Documents	Metric of the number of users who ran document jobs.
DP Percentage of Jobs with Cache Hit	Metric of the percentage of document jobs that hit a cache.
DP Percentage of Jobs with Error	Metric of the percentage of document jobs that failed.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.

Attribute or metric name	Function
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the document job.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.
Report	Indicates the reports in the document.
User	Indicates the user who ran the document job.

Document Job Step attributes and metrics

Attribute or metric name	Function
Day	Indicates the day on which the document job executed.
Document	Indicates which document was executed.
Document Job Step Sequence	Indicates the sequence number for steps in a document job.
Document Job Step Type	Indicates the type of step for a document job.
DP Average Elapsed Duration per Job (hh:mm:ss)	Metric of the average difference between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Elapsed Duration per Job (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of all document job executions.
DP Average Execution Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions.
DP Average Execution Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions.
DP Average Queue Duration per Job (hh:mm:ss)	Metric of the average duration of all document job executions waiting in the queue.
DP Average Queue Duration per Job (secs)	Metric of the average duration, in seconds, of all document job executions waiting in the queue.
DP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time (including time for prompt responses) of a document job.
DP Elapsed Duration (secs)	Metric of the average difference, in seconds, between start time and finish time (including time for prompt responses) of a document job.
DP Execution Duration (hh:mm:ss)	Metric of the duration of a document job's execution.

Attribute or metric name	Function
DP Execution Duration (secs)	Metric of the duration, in seconds, of a document job's execution.
DP Queue Duration (hh:mm:ss)	Metric of the duration of all document job executions waiting in the queue.
DP Queue Duration (secs)	Metric of the duration, in seconds, of all document job executions waiting in the queue.
Hour	Indicates the hour the document job was executed.
Metadata	Indicates the metadata storing the document.
Minute	Indicates the minute the document job was executed.
Project	Indicates the project storing the document.

Enterprise Manager Data Load attributes

Attribute name	Function
Data Load Finish Time	Displays the timestamp of the end of the data load process for the projects that are being monitored.
Data Load Project	Lists all projects that are being monitored.
Data Load Start Time	Lists the timestamp of the start of the data load process for the projects that are being monitored.
Item ID	A value of -1 indicates that it is the summary row in the EM_IS_LAST_UPDATE table for all projects in a data load. That summary row has information about how long the data load took. A value of 0 indicates it is a row with project data load details.

Inbox Message Actions attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the manipulation was started
Document	Indicates the document included in the message.
Document Job	Indicates the document job that requested the History List message manipulation.
HL Days Since Last Action: Any action	Metric of the number of days since any action was performed.
HL Days Since Last Action: Request	Metric of the number of days since the last request was made for the contents of a message.

Attribute or metric name	Function
HL Last Action Date: Any Action	Metric of the date and time of the last action performed on a message such as read, deleted, marked as read, and so on.
HL Last Action Date: Request	Metric of the date and time of the last request made for the contents of a message.
HL Number of Actions	Metric of the number of actions performed on a message.
HL Number of Actions by User	Metric of the number of actions by user performed on a message.
HL Number of Actions with Errors	Metric of the number of actions on a message that resulted in an error.
HL Number of Document Jobs	Metric of the number of document jobs that result with messages.
HL Number of Messages	Metric of the number of messages.
HL Number of Messages with Errors	Metric of the number of messages that resulted in an error.
HL Number of Messages Requested	Metric of the number of requests for the contents of a message.
HL Number of Report Jobs	Metric of the number of report jobs that result from messages.
Hour	Indicates the hour the manipulation was started on a History List message.
Inbox Action	Indicates the manipulation that was performed on a History List message.
Inbox Action Type	Indicates the type of manipulation that was performed on a History List message.
Inbox Message	Indicates the message in the History List.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the message.
Metadata	Indicates the metadata storing the message.
Minute	Indicates the minute the manipulation was started.
Project	Indicates the project storing the message.
Report	Indicates the report included in the message.
Report Job	Indicates the job ID of the report included in the message.
User	Indicates the user who manipulated the History List message.

Mobile Client attributes

Attribute name	Function
Cache Hit Indicator	Indicates whether a cache was hit during the execution and, if so, what type of cache hit.
Day	Indicates the day the action started.
Document	Identifies the document used in the request.
Execution Type Indicator	Indicates the type of report or document that initiated the execution.
Geocode	Indicates the location, in latitude and longitude form, of the user.
Hour	Indicates the hour the action started.
Intelligence Server Machine	Indicates the Intelligence Server processing the request.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute the action started.
Mobile Device Installation ID	Indicates the unique Installation ID of the mobile app.
Mobile Device Type	Indicates the type of mobile device the app is installed on, such as IPAD2, DROID, and so on.
MSTR App Version	Indicates the version of the MicroStrategy app making the request.
Network Type	Indicates the type of network used, such as 3G, WIFI, LTE, and so on.
Operating System	Indicates the operating system of the mobile device making the request.
Operating System Version	Indicates the operating system version of the mobile device making the request.
Project	Indicates the project used to initiate the request.
User	Indicates the user that initiated the request.

OLAP Services attributes and metrics

Attribute or metric name	Function
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Intelligent Cube	Indicates the Intelligent Cube that was used.
Intelligent Cube Action Duration (secs)	Metric of the duration, in seconds, for an action that was performed on the Intellgent Cube.
Intelligent Cube Action Type	Indicates the type of action taken on the Intelligent Cube such as

Attribute or metric name	Function
	cube publish, cube view hit, and so on.
Intelligent Cube Instance	Indicates the Intelligent Cube instance in memory that was used for the action.
Intelligent Cube Size (KB)	If the Intelligent Cube is published or refreshed, indicates the size, in KB, of the Intelligent Cube.
Intelligent Cube Type	Indicates the type of Intelligent Cube used, such as working set report, Report Services Base report, OLAP Cube report, and so on.
Minute	Indicates the minute on which the action was started.
Number of Dynamically Sourced Report Jobs against Intelligent Cubes	Metric of how many jobs from reports not based on Intelligent Cubes but selected by the engine to go against an Intelligent Cube because the objects on the report matched what is on the Intelligent Cube.
Number of Intelligent Cube Publishes	Metric of how many times an Intelligent Cube was published.
Number of Intelligent Cube Refreshes	Metric of how many times an Intelligent Cube was refreshed.
Number of Intelligent Cube Republishes	Metric of how many times an Intelligent Cube was republished.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Number of Users hitting Intelligent Cubes	Metric of how many users executed a report or document that used an Intelligent Cube. That is, the number of users using OLAP Services.
Number of View Report Jobs	Metric of how many actions were the result of a View Report.
Report	Indicates the report that hit the Intelligent Cube.

Performance Monitoring attributes

Attribute name	Function
Counter Category	Indicates category of the counter, such as memory, MicroStrategy server jobs, or MicroStrategy server users.
Counter Instance	Indicates the instance ID of the counter, for MicroStrategy use.
Day	Indicates the day the action was started.
Hour	Indicates the hour the action was started.
Minute	Indicates the minute the action was started.
Performance Monitor Counter	Indicates the name of the performance counter and its value type.

Prompt Answers attributes and metrics

Attribute or metric name	Function
Connection Source	Indicates the connection source to Intelligence Server.
Count of Prompt Answers	Metric of how many prompts were answered.
Day	Indicates the day the prompt was answered.
Document	Indicates the document that used the prompt.
Hour	Indicates the hour the prompt was answered.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the job.
Metadata	Indicates the metadata repository storing the prompt.
Minute	Indicates the minute the prompt was answered.
Project	Indicates the project storing the prompt.
Prompt	Indicates the prompt that was used.
Prompt Answer	Indicates the answers for the prompt in various instances.
Prompt Answer Required	Indicates whether an answer to the prompt was required.
Prompt Instance Answer	Indicates the answer of an instance of a prompt in a report job.
Prompt Location	Indicates the ID of the location in which a prompt is stored.
Prompt Location Type	Indicates the type of the object in which the prompt is stored, such as filter, template, attribute, and so on.
Prompt Title	Indicates the title of the prompt (the title the user sees when presented during job execution).
Prompt Type	Indicates what type of prompt was used, such as date, double, elements, and so on.
Report	Indicates the report that used the prompt.
Report Job	Indicates the report job that used the prompt.
RP Number of Jobs (IS_PR_ANS_FACT)	Metric of how many jobs involved a prompt.
RP Number of Jobs Containing Prompt Answer Value	Metric of how many report jobs had a specified prompt answer value.
RP Number of Jobs Not Containing Prompt Answer Value	Metric of how many report jobs did not have a specified prompt answer value.
RP Number of Jobs with Unanswered Prompts	Metric of how many report jobs had a prompt that was not answered.

Report Job attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution is ad hoc.
Cache Creation Indicator	Indicates whether an execution has created a cache.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Cancelled Indicator	Indicates whether an execution has been canceled.
Child Job Indicator	Indicates whether a job was a document dataset or a standalone report.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Database Error Indicator	Indicates whether a report request failed because of a database error.
Datamart Indicator	Indicates whether an execution created a data mart.
Day	Indicates the day on which the report was executed.
DB Instance	Indicates the database instance on which the report was executed.
Drill Indicator	Indicates whether an execution is a result of a drill.
Element Load Indicator	Indicates whether an execution is a result of an element load.
Error Indicator	Indicates whether an execution encountered an error.
Export Indicator	Indicates whether a report was exported and, if so, indicates its format.
Filter	Indicates the filter used on the report.
Hour	Indicates the hour on which the report was executed.
Intelligence Server Machine	Indicates the Intelligence Server machine that executed the report.
Metadata	Indicates the metadata repository that stores the report.
Minute	Indicates the minute on which the report execution was started.
Number of Jobs with Intelligent Cube Hit	Metric of how many job executions used an Intelligent Cube.
Project	Indicates the metadata repository that stores the report.
Prompt Indicator	Indicates whether the report execution was prompted.
Report	Indicates the ID of the report that was executed.
Report Job	Indicates an execution of a report.

Attribute or metric name	Function
RP Average Elapsed Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Elapsed Duration per Job (secs) (IS_REP_FACT)	Metric of the average difference between start time and finish time (including time for prompt responses) of all report job executions.
RP Average Execution Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average duration of all report job executions. Includes time in queue and execution for a report job.
RP Average Execution Duration per Job (secs) (IS_REP_FACT)	Metric of the average duration, in seconds, of all report job executions. Includes time in queue and execution for a report job.
RP Average Prompt Answer Time per Job (hh:mm:ss)	Metric of the average time users take to answer the set of prompts in all report jobs.
RP Average Prompt Answer Time per Job (secs)	Metric of the average time, in seconds, users take to answer the set of prompts in all report jobs.
RP Average Queue Duration per Job (hh:mm:ss) (IS_REP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Average Queue Duration per Job (secs) (IS_REP_FACT)	Metric of the average time, in seconds, report jobs waited in the Intelligence Server's queue before the report job was executed.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of a report job. Includes time for prompt responses, in queue, and execution.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Number of Ad Hoc Jobs	Metric of how many report jobs resulted from an ad hoc report creation.
RP Number of Cancelled Jobs	Metric of how many job executions were canceled.
RP Number of Drill Jobs	Metric of how many job executions resulted from a drill action.
RP Number of Jobs (IS_REP_FACT)	Metric of how many report jobs were executed.
RP Number of Jobs hitting Database	Metric of how many report jobs were executed against the database.
RP Number of Jobs w/o Cache Creation	Metric of how many report jobs were executed that did not result in creating a server cache.

Attribute or metric name	Function
RP Number of Jobs w/o Cache Hit	Metric of how many report jobs were executed that did not hit a server cache.
RP Number of Jobs w/o Element Loading	Metric of how many report jobs were executed that did not result from loading additional attribute elements.
RP Number of Jobs with Cache Creation	Metric of how many report jobs were executed that resulted in a server cache being created.
RP Number of Jobs with Cache Hit	Metric of how many report jobs were executed that hit a server cache.
RP Number of Jobs with Datamart Creation	Metric of how many report jobs were executed that resulted in a data mart being created.
RP Number of Jobs with DB Error	Metric of how many report jobs failed because of a database error.
RP Number of Jobs with Element Loading	Metric of how many report jobs were executed that resulted from loading additional attribute elements.
RP Number of Jobs with Error	Metric of how many report jobs failed because of an error.
RP Number of Jobs with Intelligent Cube Hit	Metric of how many report job executions used an Intelligent Cube.
RP Number of Jobs with Security Filter	Metric of how many report job executions used a security filter.
RP Number of Jobs with SQL Execution	Metric of how many report jobs executed SQL statements.
RP number of Narrowcast Server jobs	Metric of how many report job executions were run through MicroStrategy Narrowcast Server.
RP Number of Prompted Jobs	Metric of how many report job executions included a prompt.
RP Number of Report Jobs from Document Execution	Metric of how many report jobs executed as a result of a document execution.
RP Number of Result Rows	Metric of how many result rows were returned from a report execution.
RP Number of Scheduled Jobs	Metric of how many report jobs were scheduled.
RP Number of Users who ran reports	Metric of how many distinct users ran report jobs.
RP Prompt Answer Duration (hh:mm:ss)	Metric of the how long users take to answer the set of prompts in report jobs.
RP Prompt Answer Duration (secs)	Metric of the how long, in seconds, users take to answer the set of prompts in report jobs.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
Schedule	Indicates the schedule that began the report execution.
Schedule Indicator	Indicates whether the report execution was scheduled.
Security Filter	Indicates the security filter used in the report execution.
Security Filter Indicator	Indicates whether a security filter was used in the report execution.
SQL Execution Indicator	Indicates that SQL was executed during report execution.
Template	Indicates the report template that was used.
User	Indicates the user that ran the report.

Report Job SQL Pass attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether the execution was ad hoc.
Connection Source	Indicates the connection source to Intelligence Server.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Metadata	Indicates the metadata repository storing the report or document.
Minute	Indicates the minute in which the report job was started.
Project	Indicates the project storing the report or document.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job SQL Pass	Indicates the SQL statement that was executed during the SQL pass.
Report Job SQL Pass Type	Indicates the type of SQL statement that was executed in this SQL pass. Examples are SQL select, SQL insert, SQL create and such.
RP Execution Duration (hh:mm:ss)	Metric of the duration of a report job's execution. Includes database execution time.
RP Execution Duration (secs)	Metric of the duration, in seconds, of a report job's execution. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.

Attribute or metric name	Function
RP Number of DB Tables Accessed	Metric of how many database tables were accessed in a report job execution.
RP SQL Size	Metric of how large, in bytes, the SQL was for a report job.

Report Job Steps attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Cache Hit Indicator	Indicates whether an execution has hit a cache.
Connection Source	Indicates the connection source to Intelligence Server.
Cube Hit Indicator	Indicates whether an execution hit an intelligent cube or database.
Day	Indicates the day in which the job was executed.
Hour	Indicates the hour in which the report job was executed.
Minute	Indicates the minute in which the report job was started.
Report	Indicates the report that was executed.
Report Job	Indicates an execution of a report.
Report Job Step Sequence	Indicates the sequence number in the series of execution steps a report job passes through in the Intelligence Server.
Report Job Step Type	Indicates the type of step for a report job. Examples are SQL generation, SQL execution, Analytical Engine, Resolution Server, element request, update Intelligent Cube, and so on.
RP Average CPU Execution Duration per Job (msecs) (IS_REP_STEP_FACT)	Metric of the average duration, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Average Elapsed Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Average Query Engine Execution Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time, in seconds, the Query Engine takes to process a report job.
RP Average Queue Duration per Job (secs) (IS_REP_STEP_FACT)	Metric of the average time report jobs waited in the Intelligence Server's queue before the report job was executed.

Attribute or metric name	Function
RP CPU Duration (msec)	Metric of how long, in milliseconds, a report job execution takes in the Intelligence Server CPU.
RP Elapsed Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes time for prompt responses.
RP Elapsed Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes time for prompt responses.
RP Execution Duration (hh:mm:ss)	Metric of the difference between start time and finish time of report job executions. Includes database execution time.
RP Execution Duration (secs)	Metric of the difference, in seconds, between start time and finish time of report job executions. Includes database execution time.
RP Last Execution Finish Timestamp	Metric of the finish timestamp when the report job was last executed.
RP Last Execution Start Timestamp	Metric of the start timestamp when the report job was last executed.
RP Number of Jobs (IS_REP_STEP_FACT)	Metric of how many report jobs were executed.
RP Query Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the Query Engine took to execute SQL for a report job.
RP Query Engine Duration (secs) (IS_REP_STEP_FACT)	Metric of the time, in seconds, the Query Engine takes to execute SQL for a report job.
RP Queue Duration (hh:mm:ss)	Metric of how long a report job waited in the Intelligence Server's queue before the report job was executed.
RP Queue Duration (secs)	Metric of how long, in seconds, a report job waited in the Intelligence Server's queue before the report job was executed.
RP SQL Engine Duration (hh:mm:ss) (IS_REP_STEP_FACT)	Metric of how long the SQL Engine took to generate SQL for a report job.

Report Job Tables/Columns Accessed attributes and metrics

Attribute or metric name	Function
Ad Hoc Indicator	Indicates whether an execution was ad hoc.
Column	Indicates the column that was accessed.
Connection Source	Indicates the connection source to Intelligence Server.

Attribute or metric name	Function
Day	Indicates the day on which the table column was accessed.
DB Table	Indicates the table in the database storing the column that was accessed.
Hour	Indicates the hour on which the table column was accessed.
Minute	Indicates the minute on which the table column was accessed.
Report	Indicates the report that accessed the table column.
Report Job	Indicates which execution of a report accessed the table column.
RP Number of Jobs (IS_REP_COL_FACT)	Metric of how many report jobs accessed the database column or table. The Warehouse Tables Accessed report uses this metric.
SQL Clause Type	Indicates which type of SQL clause was used to access the table column.

Schema Objects attributes

Attribute name	Function
Attribute	Lists all attributes in projects that are set up to be monitored by Enterprise Manager.
Attribute Form	Lists all attribute forms in projects that are set up to be monitored by Enterprise Manager.
Column	Lists all columns in projects that are set up to be monitored by Enterprise Manager.
DB Table	Lists all physical tables in the data warehouse that are set up to be monitored by Enterprise Manager.
Fact	Lists all facts in projects that are set up to be monitored by Enterprise Manager.
Hierarchy	Lists all hierarchies in projects that are set up to be monitored by Enterprise Manager
Table	Lists all logical tables in projects that are set up to be monitored by Enterprise Manager.
Transformation	Lists all transformations in projects that are set up to be monitored by Enterprise Manager.

Server Machines attributes

Attribute name	Function
Client Machine	Lists all machines that have had users connect to the Intelligence Server.
Intelligence Server Cluster	Lists the cluster of Intelligence Servers.
Intelligence Server Machine	Lists all machines that have logged statistics as an Intelligence Server.
Web Server Machine	Lists all machines used as web servers.

Session attributes and metrics

Attribute or metric name	Function
Avg. Connection Duration (hh:mm:ss)	Metric of the average time connections to an Intelligence Server last.
Avg. Connection Duration (secs)	Metric of the average time, in seconds, connections to an Intelligence Server last.
Connection Duration (hh:mm:ss)	Metric of the time a connection to an Intelligence Server lasts.
Connection Duration (secs)	Metric of the time, in seconds, a connection to an Intelligence Server lasts.
Connection Source	Lists all connection sources to Intelligence Server.
Number of Sessions (Report Level)	Metric of how many sessions were connected to an Intelligence Server. Usually reported with a date and time attribute.
Number of Users Logged In (Report Level)	Metric of how many distinct users were connected to an Intelligence Server. Usually reported with a date and time attribute.
Session	Indicates a user connection to an Intelligence Server.

8

MicroStrategy Web Cookies

This chapter provides detailed information for all cookies used in MicroStrategy Web.

In addition to listing the different cookies used in the application, this chapter describes the potential consequences for each cookie if it gets lost during the HTTP request process.

Session information

`MSTRSsn_<ServerName>_<ProjectName>_<PortNumber>`

This cookie saves relevant information regarding a specific session with an Intelligence Server. This is a temporary cookie that is destroyed when the browser is closed or the client's Web session expires.

If, for any reason, this cookie is not present at request time, the application first redirects the execution to the login page, sign the user back in, and then redirect the flow of the application to the page originally requested. When the flow is redirected to the login page, if the user logged in using standard authentication, the login page prompts the user for his or her password. If the user logged in using either Windows or Guest authentication, the application automatically logs the user in; the user will not even notice it. This happens because the session ID is lost.

Cookie subkeys

Subkey	Purpose
Tkn	Stores the Intelligence Server session ID.
PWD	Stores the password used to log in. An Administrator Preference determines whether the password is encrypted or not (not encrypted by default). The password is saved here if the user has not chosen to save the password on the Web browser.
NT	Indicates whether the user logged in with Windows authentication.

Subkey	Purpose
	0 means the user was not logged in using Windows credentials. 1 means the user was logged in using Windows credentials.
Gst	Indicates whether the user logged in as Guest. 0 means the user was not logged as Guest. 1 means the user was logged as Guest.
Prv1	Stores the first set of User Privileges as a hexadecimal value (denoted by &H). Possible values are as follows: PRIVILEGE_WEBADMINISTRATOR = &H1 PRIVILEGE_WEBUSER = &H2 PRIVILEGE_WEBVIEWHISTORYLIST = &H4 PRIVILEGE_WEBREPORTMANIPULATIONS = &H8 PRIVILEGE_WEBCREATENEWREPORT = &H10 PRIVILEGE_WEBOBJECTSEARCH = &H20 PRIVILEGE_WEBCHANGEUSEROPTIONS = &H40 PRIVILEGE_WEBSAVEREPORT = &H80 PRIVILEGE_WEBDRILLANYWHERE = &H100 PRIVILEGE_WEBEXPORT = &H200 PRIVILEGE_WEBPRINTMODE = &H400 PRIVILEGE_WEBDELETE = &H800 PRIVILEGE_WEBPUBLISH = &H1000 PRIVILEGE_WEBREPORTDETAILS = &H2000 PRIVILEGE_WEBREPORTSQL = &H4000 PRIVILEGE_WEBADDHISTORYLIST = &H00008000 PRIVILEGE_WEBCHANGEVIEWMODE = &H10000 PRIVILEGE_WEBDRILL = &H20000 PRIVILEGE_WEBDRILLONMETRICS = &H40000 PRIVILEGE_WEBCHANGESTYLE = &H80000 PRIVILEGE_WEBSCHEDULING = &H100000 PRIVILEGE_WEBSIMULTANEOUSEXECUTION = &H200000 PRIVILEGE_WEBSORT = &H400000 PRIVILEGE_WEBSWITCHPAGEBY = &H800000 PRIVILEGE_WEBSAVETEMPLATEFILTER = &H1000000 PRIVILEGE_WEBFILTERONSELECTION = &H2000000 PRIVILEGE_WEBUSEREPORTFILTEREDITOR = &H4000000

Subkey	Purpose
	PRIVILEGE_WEBCREATEDERIVEDMETRICS = &H8000000 PRIVILEGE_WEBMODIFYSUBTOTALS = &H10000000 PRIVILEGE_WEBUSEREPORTOBJECTSWINDOW = &H20000000
Prv2	Stores the second set of User Privileges as a hexadecimal value (denoted by &H). Possible values are as follows: PRIVILEGE_WEBFORMATTINGEDITOR = &H40000001 PRIVILEGE_WEBSCHEDULEEMAIL = &H40000002 PRIVILEGE_WEBSENDNOW = &H40000004 PRIVILEGE_WEBMODIFYREPORTLIST = &H40000008 PRIVILEGE_WEBUSEDESIGNMODE = &H40000010 PRIVILEGE_WEBALIASOBJECTS = &H40000020 PRIVILEGE_WEBCONFIGURETOOLBARS = &H40000040 PRIVILEGE_WEBUSEQUERYFILTEREDITOR = &H40000080 PRIVILEGE_WEBREEXECUTEREPORTAGAINSTWH = &H40000100 PRIVILEGE_WEBSIMPLEGRAPHFORMATTING = &H40000200 PRIVILEGE_WEBUSELOCKEDHEADERS = &H40000400 PRIVILEGE_WEBSETCOLUMNWIDTHS = &H40000800
Lng	Indicates the locale ID associated with the Intelligence Server session.
ReturnURL	Stores the URL Query String of the last report, document, or folder visited or executed.
ReturnName	Stores the metadata object name of the last report, document, or folder visited or executed.
StartPageURL	Stores the URL Query String of the default start page for the Intelligence Server project.
StartPageName	Stores the Web Application Page or Feature name of the default start page for the project. For example, Shared Reports, My Reports, and so on.
WrkSet	Indicates if the user has privileges to use the Working Set.
Template	Stores the `DssObjectId` of the last template chosen by the user (by clicking on it while browsing folders inside the Web application).
Filter	Stores the `DssObjectId` of the last filter chosen by the user (by clicking on it while browsing folders inside the Web application).
ShowFilterTemplate	Indicates whether to allow the user to execute filter and template combinations.
Categories	Indicates the number of categories for the graph mode.

Subkey	Purpose
Series	Indicates the number of series for the graph mode.
ShRptsID	The folder ID that used for the “Shared Reports” folder.
ShRprtsName	The name of the folder used for the “Shared Reports” folder.
Utf	Indicates if the client is in Unicode mode.
Ver	Stores the Intelligence Server version number.
CurrProjID	Stores the current project ID.

Default user name

`MSTRDfltUsrNm`

Stores the last user ID logged successfully into the application. This is a permanent cookie. This cookie is only used at login time to preset the user ID on the login page. If the cookie is not present, the user ID text box on the login page could be empty.

Project information

`MSTRPrj_<Server_name>_<Project_Name>`

This cookie stores information relevant to a specific project. This is a permanent cookie. Information stored in this cookie is available even if the web browser is closed or the client's Web session expires.

If this cookie is not present at request time, the application automatically logs the user in again, using the information contained in the URL Query String and the Connection Info cookie. This happens because the application believes the user is trying to connect to a new project even if this is not true. Since the User ID is saved in this cookie, if the cookie is not present, the user ID label that shows up on the Developer page is lost. The user preferences are also saved in this cookie so they too will be lost.

Cookie subkeys

Subkey	Purpose
Uid	Stores the user ID last used to log in to this project.
PWD	Stores the password used to log in if the user decided to save the password in the Web browser. The value of this cookie is never encrypted.
Disp_Mode	Indicates whether or not the user had the outline mode option in the report page selected the last time he used the application.

Subkey	Purpose
aa ... io	This set of cookies—from the `aa` subkey to the `io` subkey—represents the user preferences. Each user preference has been assigned one subkey. For a complete list of preferences and their codes, see *Preferences, page 874*.

Current language

`Lng`

Indicates the locale ID used in the current or last visited project. This is a temporary cookie. This cookie is created at login time. If the cookie is lost in the middle of a session, the application recovers this information from the information included in the session information cookie.

GUI settings

`MstrWeb`

This cookie stores information specific to the look and feel of the application. This is a temporary cookie. If the cookie is not present at request time, the values it holds take the default values.

Cookie subkeys

Subkey	Purpose
Toolbar	Indicates whether the left toolbar is visible or not: • 0 means the toolbar is not shown to the user. • 1 means the toolbar is shown to the user.
HelpSection	Indicates whether the help section on the left toolbar is visible or not: • 0 means the help section is not shown to the user. • 1 means the help section is shown to the user.

Personal autostyle information

`PersonalAutoStyleInfo`

Stores the ID and name of autostyles that are stored in the 'My objects' folder. This cookie is created when a user logs into the MicroStrategy Web application.

System autostyle information

`SystemAutoStyleInfo`

Stores the ID and name of default autostyles that are created when you upgrade a project to version 7.2.x and that are stored in the 'Autostyles' folder. This cookie is created when a user logs into the MicroStrategy Web application.

Connection information

`ConnectionInfo`

This cookie stores the information relevant to the currently connected Intelligence Server. This is a temporary cookie. If the information in this cookie is lost during a request, the application logs the user in again with the information contained on the URL Query String. This happens because the application believes the user is trying to connect to a new project even if this is not true.

Cookie subkeys

Subkey	Purpose
CurrServer	Stores the name of the Intelligence Server machine.
CurrProject	Stores the name of the Intelligence Server's project.
CurrPort	Stores the Intelligence Server port number.

Available projects information

`XLgn`

This cookie is used for caching the information about the Intelligence Servers to which the Web Server can potentially connect and the projects each Intelligence Server contains. This is a temporary cookie. The subkeys are a set of four different cookies. If this cookie is not received at request time, it is created from the Clusters collection in the XML API.

Cookie subkeys

Subkey	Purpose
P<n>SERVER	Stores the Intelligence Server machine name.

Subkey	Purpose
P<n>PROJECT	Stores the Intelligence Server's project name.
P<n>PORT	Stores the Intelligence Server's port number.
P<n>ALIAS	Stores the project alias of the project. This alias is set via the Preferences page.

Global user preferences

`UsrOpt`

This cookie stores the user Preferences common to all the projects and Intelligence Servers. This is a permanent cookie. This cookie is read at login time. If the cookie is not received, the application assumes global user preferences have been saved when caching the preferences.

Cookie subkeys

Subkey	Purpose
aa ... io	This set of cookies—from the `aa` subkey to the `io` subkey—represents the user preferences. Each user preference has been assigned one subkey. For a complete list of preferences and their codes, see *Preferences, page 874*.

Cached preferences

`CurrUsrOpt`

When the user enters the application and logs in to a specific project, all the applicable preferences, either from the Administrator Preferences, Project Defaults, or the User Preferences are combined and cached in this cookie. This is a temporary cookie. The application assumes all cookies are cached at login time. If the information contained in this cookie is lost, the application might behave unpredictably depending on the preference being read. All check box preferences behave as FALSE or cleared, numeric preferences might raise an error.

Cookie subkeys

Subkey	Purpose
aa ... io	This set of cookies—from the `aa` subkey to the `io` subkey—represents the user preferences. Each user preference has been assigned one subkey. For a complete list of preferences and their codes, see *Preferences, page 874*.

Preferences

For reference, this section lists the different preferences and their corresponding codes.

Code	Preference	Code	Preference	Code	Preference
aa	GRID_STYLE_OPTION	da	DROPDOWN_START_PAGE_URL_OPTION	fv	EXPORT_PDF_NUM_COLS_PER_PAGE_OPTION
ab	USE_DEFAULT_GRID_STYLE_OPTION	db	CURRENT_START_PAGE_NAME_OPTION	fw	PROMPT_ON_EXPORT_PDF_OPTION
ac	GRID_ROWS_OPTION	dc	CURRENT_START_PAGE_URL_OPTION	fx	ACCESSIBILITY_OPTION
ad	GRID_COLUMNS_OPTION	dd	NEW_START_PAGE_NAME_OPTION	fy	PAGE_BY_EDITOR_OPTION
ae	LOCALE_OPTION	de	NEW_START_PAGE_URL_OPTION	fz	FILTER_EDITOR_OPTION
af	EXPORT_FORMAT_OPTION	df	ACTUAL_START_PAGE_URL_OPTION	ga	OB_SEARCH_ID_OPTION
ag	USER_HEADER_OPTION	dg	ACTUAL_START_PAGE_NAME_OPTION	gb	FORMULA_BAR_OPTION
ah	USER_FOOTER_OPTION	dh	PROMPT_ADMIN_CANCEL_JOBS_OPTION	gc	REPORT_TOOLBAR_OPTION
ai	PRINT_ROWS_OPTION	di	PROMPT_ADMIN_DELETE_JOBS_OPTION	gd	WORKING_SET_OPTION
aj	PRINT_COLUMNS_OPTION	dj	ALLOW_USER_EXPORT_TEMP_FILES_OPTION	ge	EXPORT_PDF_MAX_ROWS_PER_PAGE_OPTION
ak	MAX_PRINT_ROWS_OPTION	dk	ALLOW_USER_USE_DHTML_OPTION	gf	EXPORT_PDF_MAX_COLS_PER_PAGE_OPTION
al	MAX_PRINT_COLUMNS_OPTION	dl	KEEP_PARENT_OPTION	gg	PRINT_HEADER_FOOTER_OPTION
am	START_PAGE_OPTION	dm	USE_EXPORT_TEMP_FILES_OPTION	gh	TOP_EDITOR_OPTION
an	CANCEL_JOBS_OPTION	do	WORKING_SET_SIZE_OPTION	gi	VIEW_FILTER_EDITOR_OPTION
ao	DELETE_JOBS_OPTION	dp	DRILL_PRIVILEGE_OPTION	gj	EXPORT_ALL_HEADERS_AS_TEXT_OPTION

Code	Preference	Code	Preference	Code	Preference
ap	SAVE_PWD_OPTION	dq	USE_DHTML_ PROMPTS_OPTION	gk	ENCRYPT_ CONNECTION_OPTION
aq	AUTHENTICATION_ OPTION	dr	USE_DHTML_VALUE_ OPTION	gl	OB_FOLDER_ID_ OPTION
ar	INBOX_EXPIRES_ OPTION	ds	SEARCH_WORKING_ SET_SIZE_OPTION	gm	OB_SEARCH_TEXT_ OPTION
as	INBOX_MAX_JOBS_ OPTION	dt	GRAPH_FORMAT_ OPTION	gn	OB_START_OBJECT_ OPTION
at	INBOX_MAX_SIZE_ OPTION	du	PREVIEW_NEW_ WINDOW_OPTION	go	FILTER_SHOW_EDIT_ OPTION
au	GRAPH_SIZE_OPTION	dv	EXPORT_VALUES_AS_ TEXT_OPTION	gp	FILTER_AUTO_APPLY_ OPTION
av	GRAPH_HEIGHT_ OPTION	dw	REFRESH_METHOD_ OPTION	gq	FORMULABAR_OPTION
aw	GRAPH_WIDTH_ OPTION	dx	PRINT_GRID_AND_ GRAPH_TOGETHER_ OPTION	gr	PRINT_HEADER_LEFT_ OPTION
ax	OBJECT_BROWSING_ OPTION	dy	REPEAT_COLHEADERS_ OPTION	gs	PRINT_HEADER_ CENTER_OPTION
ay	EXPORT_NEW_ WINDOW_OPTION	dz	EXPORT_ROWS_ FORMATTING_OPTION	gt	Not used; HTML reserved character
az	PAGE_NUMBERING_ OPTION	ea	EXPORT_COLS_ SPREADSHEET_ OPTION	gu	PRINT_HEADER_RIGHT_ OPTION
ba	PAGE_LAYOUT_ OPTION	eb	EXPORT_GRAPH_ FORMAT_OPTION	gv	PRINT_FOOTER_LEFT_ OPTION
bb	ADMIN_HEADER_ OPTION	ec	EXPORT_GRAPH_AND_ GRID_FORMAT_ OPTION	gw	PRINT_FOOTER_ CENTER_OPTION
bc	ADMIN_FOOTER_ OPTION	ed	POP_UP_DRILL_ OPTION	gx	PRINT_FOOTER_RIGHT_ OPTION
bd	MAX_EXPORT_COLS_ OPTION	ef	PROMPTS_ON_ONE_ PAGE_OPTION	gy	FORMAT_SEARCH_ID_ OPTION
bf	GRID_CSS_OPTION	eg	REQUIRED_PROMPTS_ FIRST_OPTION	gz	FORMAT_BAR_OPTION
bg	MAX_EXPORT_ROWS_ OPTION	eh	GRAPH_CATEGORIES_ AND_SERIES_FROM_ DESKTOP_OPTION	ha	DD_MENU_OPTION
bh	MAX_EXPORT_ROWS_ TEXT_OPTION	ei	GRAPH_CATEGORIES_ OPTION	hb	UNICODE_OPTION

Code	Preference	Code	Preference	Code	Preference
bi	DEFAULT_ AUTHENTICATION_ OPTION	ej	GRAPH_SERIES_ OPTION	hc	REPORT_BAR_OPTION
bk	PIVOT_MODE_OPTION	ek	EXPORT_PLAINTEXT_ DELIMITER_OPTION	hd	CHANGE_EXPIRED_ PWD_OPTION
bm	INBOX_SORT_OPTION	el	SEARCH_OBJECTS_ OPTION	he	PRINT_CELLS_PER_ BLOCK_OPTION
bn	CANCEL_JOBS_ PROMPT_OPTION	em	DOCUMENT_EXPORT_ FORMAT_OPTION	hf	PRINT_GRAPHS_PER_ BLOCK_OPTION
bo	DELETE_JOBS_ PROMPT_OPTION	en	ALLOW_USER_SET_ GRAPH_SETTINGS_ OPTION	hg	PRINT_MARGIN_LEFT_ OPTION
bp	USE_SECURITY_ PLUGIN_OPTION	eo	REPORT_TAB_OPTION	hh	PRINT_MARGIN_RIGHT_ OPTION
bq	SECURITY_PLUGIN_ CLASS_OPTION	ep	REUSE_MESSAGE_ FOR_SCHEDULED_ REPORTS_OPTION	hi	PRINT_MARGIN_TOP_ OPTION
br	SECURITY_PLUGIN_ FREQ_OPTION	eq	CONNECTED_ PROJECT_OPTION	hj	PRINT_MARGIN_ BOTTOM_OPTION
bs	PROMPT_ON_PRINT_ OPTION	er	SUBSCRIPTION_VIEW_ MODE_OPTION	hk	PRINT_HEADER_SIZE_ OPTION
bt	PROMPT_ON_EXPORT_ OPTION	es	ATTRIBUTE_DISPLAY_ IN_GRIDS_OPTION	hl	PRINT_FOOTER_SIZE_ OPTION
bu	EXECUTION_MODE_ OPTION	et	ALLOWED_FILE_ EXTENSION_OPTION	hm	PRINT_POPUP_PRINT_ DIALOG_OPTION
bw	EXECUTION_WAIT_ TIME_OPTION	eu	MAXIMUM_FILE_SIZE_ TO_UPLOAD_OPTION	hn	PRINT_COVER_PAGE_ OPTION
by	PRINT_FILTER_ DETAILS_OPTION	ev	WARNING_LEVEL_ OPTION	ho	PRINT_FIT_ROW_TO_ PAGE_OPTION
bz	EXPORT_FILTER_ DETAILS_OPTION	ew	TRACE_LEVEL_OPTION	hp	PRINT_FIT_COL_TO_ PAGE_OPTION
ca	ELE_PROMPT_BLOCK_ COUNT_OPTION	ex	ERROR_LEVEL_ OPTION	hq	PRINT_SCALING_ OPTION
cb	EXPORT_SECTION_ OPTION	ey	MAX_ELEMENTS_TO_ IMPORT_OPTION	hr	PRINT_SHRINK_FONT_ OPTION
cc	SHOW_DATA_OPTION	ez	SUBSCRIPTION_SORT_ OPTION	hs	SERVER_LOCALE_ OPTION
cd	SHOW_TOOLBAR_ OPTION	fa	AUTOMATIC_PAGE_BY_ OPTION	ht	HIDE_MYREPORTS_ OPTION

Code	Preference	Code	Preference	Code	Preference
ce	SHOW_HELP_OPTION	fb	USE_TIMESTAMP_ VALUE_OPTION	hu	IFRAME_YES_NO_ OPTION
cf	WAIT_TIME_IN_WAIT_ PAGE_OPTION	fc	FONT_FAMILY_OPTION	hv	ALLOW_SEAMLESS_ LOGIN_OPTION
cg	ADMIN_DELETE_JOBS_ OPTION	fd	SORT_METRICS_PAGE_ BY_OPTION	hx	SECURE_OPTION
ch	OBJ_PROMPT_BLOCK_ COUNT_OPTION	fe	OVERWRITE_CSS_ FONT_OPTION	hy	WORDWRAP_ ATTRIBUTES_OPTION
ci	FORM_METHOD_ OPTION	ff	FILTER_TEMPLATE_ OPTION	hz	ENABLE_PDF_EXPORT_ OPTION
cj	DEFAULT_START_ DOCUMENT_OPTION	fg	SUBSCRIPTION_ OPTION	ia	PRINT_EXPAND_ PAGEBY_OPTION
ck	ADMIN_CANCEL_JOBS_ OPTION	fh	FILTER_ON_ SELECTION_OPTION	ib	ENABLE_LOCK_ HEADERS_OPTION
cl	ICON_VIEW_MODE_ OPTION	fi	PAGE_BY_DHTML_ OPTION	ic	ENABLE_ADD_ EMBEDDED_FILTER_ OPTION
cm	MAX_SEARCH_ RESULTS_OPTION	fj	START_PAGE_LAYOUT_ OPTION	id	SHARED_REPORTS_ FOLDER_ID_NAME_ OPTION
cn	SEARCH_TIMEOUT_ OPTION	fk	START_PAGE_NAME_ OPTION	ie	ENABLE_HIDDEN_ FORMS_ADV_SORT_ OPTION
co	DEFAULT_OBJECT_ NAME_OPTION	fl	START_PAGE_ COMPONENTS_OPTION	if	SAVE_REPORT_ SETTINGS_OPTION
cp	MAX_PROJECT_NAME_ OPTION	fm	START_PAGE_SAVED_ OPTION	ig	DISABLE_RMC_DRILL_ OPTION
cq	PROJECT_ALIAS_ OPTION	fn	EXPORT_ALL_VALUES_ AS_TEXT_OPTION	ih	SHOW_EVAL_LEVEL_ VIEW_FILTER_OPTION
cr	MAX_PROJECT_TABS_ OPTION	fo	EXPORT_PDF_PAGE_ ORIENTATION_OPTION	ii	ENABLE_COLUMN_ WIDTHS_OPTION
cs	SHOW_TAB_OPTION	fp	EXPORT_PDF_PAPER_ SIZE_OPTION	ij	HIGH_SECURY_CHECK_ OPTION
ct	ADD_TO_MY_HISTORY_ LIST_OPTION	fq	EXPORT_PDF_SHRINK_ GRAPH_OPTION	ik	TIME_ZONE_OPTION
cv	ORIENTATION_ PREVIEW_OPTION	fr	EXPORT_PDF_GRID_ GRAPH_SAME_PAGE_ OPTION	il	OFFICE_VERSION_ OPTION
cw	PAPER_SIZE_PREVIEW_ OPTION	fs	EXPORT_PDF_MAX_ ROWS_OPTION	im	EXPORT_PDF_ NUMBER_CELLS_

Code	Preference	Code	Preference	Code	Preference
					OPTION
cx	ADMIN_CONTACT_ INFO_OPTION	ft	EXPORT_PDF_MAX_ COLS_OPTION	in	DEFAULT_HEADER_ WIDTH_OPTION
cz	START_PAGE_RADIO_ OPTION	fu	EXPORT_PDF_NUM_ ROWS_PER_PAGE_ OPTION	io	DEFAULT_GRID_WIDTH_ OPTION

INDEX

A

accessing VLDB Properties Editor _19

Administration privileges _330

advanced VLDB properties _22

Analyst group privileges _310

Analyst privileges _326

  privileges common to Web Reporter _323

Architect privileges _329

attribute

  dynamic sourcing, disabling for _41

  inner join in dynamic sourcing _41

  outer join in dynamic sourcing _41

attribute element in dynamic sourcing _41

attribute form qualification in dynamic sourcing _45

B

BigInt support _201

C

cache for internationalization _231

changing VLDB properties _22, 27

  warehouse database instance _26

Command Manager privileges _330

Command Manager Runtime _334

  executing a script with _334

  syntax _336

common privileges for Web Reporter and Analyst _323

cookie

  available projects information _872

  cached preferences _873

  connection information _872

  current language _871

  default user name _870

  global user preferences _873

  GUI settings _871

  preferences _874

  project information _870

session information _867

creating a VLDB settings report _21

D

data mart that supports BigInt _201

data warehouse in Enterprise Manager _415

database

default VLDB properties _225

database gateway _161

default

resetting VLDB properties _23

VLDB properties for database _225

Developer group privileges _310

Developer privileges _327

displaying advanced VLDB properties _23

Distribution Services privileges _325

dynamic aggregation level _113

dynamic sourcing

filter in _45

inner join in _41, 44

metric in _44

null value in _41

outer join in _41, 44

troubleshooting _41

VLDB properties _38, 41, 45

E

Enterprise Manager

attribute description _457

data warehouse table _415

fact table _416

lookup table _445

metadata table _454, 456

relationship table _453, 455

transformation table _451

evaluation order VLDB properties _32

Everyone group privileges _310

Export Engine VLDB properties _46

F

filter

dynamic sourcing _45

VLDB properties _33, 182-183, 207

Freeform SQL

VLDB properties _47

Freeform SQL VLDB properties _47

G

governing VLDB properties _49

H

hidden VLDB properties _22

hierarchy, unbalanced or ragged _20

I

indexing VLDB properties _53

inner join

for attributes in dynamic sourcing _41

for metrics in dynamic sourcing _44

Integrity Manager privileges _330

Intelligence Server statistics tables _337

internationalization _230

caching _231

J

Join 92 *73*

M

MDX cube VLDB properties *20*

MDX VLDB properties *85*

metric

- dynamic aggregation VLDB property *113*
- dynamic sourcing *44*
- inheriting MDX cube source format *90*
- inner join in dynamic sourcing *44*
- outer join in dynamic sourcing *44*
- VLDB properties *34, 97, 183*

MicroStrategy Architect group privileges *310*

MicroStrategy Mobile

- privileges *324*

MicroStrategy MultiSource Option

- incremental data transfer *160*

MicroStrategy Office

- privileges *324*

MicroStrategy Web Analyst group privileges *311*

MicroStrategy Web Professional group privileges *311*

MicroStrategy Web Reporter group privileges *311*

modifying VLDB properties *26-27*

MultiSource Option

- enabling *161*
- parallel query execution *163*
- privileges *326*

N

Narrowcast System Administrators group privileges *312*

null value

- dynamic sourcing and *41, 44*
- troubleshooting *41, 44*

O

Object Manager privileges *329*

object name personalization *280*

order of precedence of VLDB properties *18*

outer join for dynamic sourcing *41, 44*

P

parallel query execution *162*

- maximum parallel queries per report *160*
- using with MultiSource Option *163*

parallel SQL execution

- improvement estimate *165, 224*

power user privileges *308*

pre/post statement VLDB properties *117*

privileges *315*

- Administration *330*
- Analyst *326*
- Analyst group *310*
- Architect *329*
- Command Manager *330*
- default security role *308*
- default user group *309*

Developer _327
Developer group _310
Distribution Services _325
Everyone group _310
Integrity Manager _330
list of all privileges _315
MicroStrategy Architect group _310
MicroStrategy Mobile _324
MicroStrategy Office _324
MicroStrategy Web Analyst group _311
MicroStrategy Web Professional group _311
MicroStrategy Web Reporter group _311
MultiSource Option _326
Narrowcast System Administrators group _312
Object Manager _329
power user _308
project bulk administrator _308
project operations administrator _308
project operations monitor _308
project resource settings administrator _309
project security administrator _309
Server Configuration Administrators group _312
Server Operations Administrators group _312
Server Operations Monitors group _312
Server Resource Settings Administrators group _313
Server Security Administrators group _313
System Administrators group _313
System Monitors group _312
User Administrators group _313
Web Analyst _317
Web Professional _320
Web Reporter _316
project bulk administrator privileges _308
project operations administrator privileges _308
project operations monitor privileges _308
project resource settings administrator privileges _309
project security administrator privileges _309
properties, displaying advanced _23

Q

query optimization VLDB properties _135

R

ragged hierarchy _20

S

security role default privileges _308
select/insert VLDB properties _185
Server Configuration Administrators group privileges _312
Server Operations Administrators group privileges _312
Server Operations Monitors group privileges _312

Server Resource Settings Administrators group privileges *313*

Server Security Administrators group privileges *313*

statistics tables *337*

support. See technical support. *16*

System Administrators group privileges *313*

System Monitors group privileges *312*

T

table VLDB properties *207*

technical support *16*

Transaction Services

- VLDB properties *48*

translating

- already translated projects *243*
- creating translator role *274*
- project initially *238*

translator roles *274*

troubleshooting

- dynamic sourcing *41*
- null value *41, 44*

U

unbalanced hierarchy *20*

upgrading VLDB properties for a database type *25*

User Administrators group privileges *313*

V

viewing VLDB properties *22*

VLDB properties *17*

- advanced *22*
- apply filter options *29*
- changing *22, 27*
- custom group display *30*
- database defaults *225*
- dynamic sourcing *38*
- dynamic sourcing, in *41, 45*
- evaluation ordering *32*
- Export Engine *46*
- filter *33, 182-183, 207*
- Freeform SQL *47*
- governing *49*
- ignore empty result for Freeform SQL *47*
- indexing *53*
- joint element lists *30*
- level of dynamic aggregation *113*
- MDX *85*
- metric *34, 97, 183*
- modifying *27*
- modifying for a warehouse database instance *26*
- order of precedence *18*
- pre/post statements *117*
- query optimizations *135*
- select/insert *185*
- setting to default *23*
- settings report *20*
- tables *207*
- upgrading for a database type *25*
- viewing *22*

XQuery Success Code_48
VLDB Properties Editor_19
VLDB settings_17
creating report_21
report_20

W

Web Analyst privileges_317
Web Professional privileges_320
Web Reporter privileges_316
privileges common to Analyst_323