Twelfth Edition, June 2011, version 9.2.1

To ensure that you are using the documentation that corresponds to the software you are licensed to use, compare this version number with the software version shown in “About MicroStrategy...” in the Help menu of your software.

Document number: 09300921
Copyright © 2011 by MicroStrategy Incorporated. All rights reserved.

If you have not executed a written or electronic agreement with MicroStrategy or any authorized MicroStrategy distributor, the following terms apply:

This software and documentation are the proprietary and confidential information of MicroStrategy Incorporated and may not be provided to any other person. Copyright © 2001-2011 by MicroStrategy Incorporated. All rights reserved.

This SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" AND WITHOUT EXPRESS OR LIMITED WARRANTY OF ANY KIND BY EITHER MICROSTRATEGY INCORPORATED OR ANYONE WHO HAS BEEN INVOLVED IN THE CREATION, PRODUCTION, OR DISTRIBUTION OF THE SOFTWARE OR DOCUMENTATION, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, GOOD TITLE AND NON-INFRINGEMENT, QUALITY OR ACCURACY. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE SOFTWARE AND DOCUMENTATION IS WITH YOU. SHOULD THE SOFTWARE OR DOCUMENTATION PROVE DEFECTIVE, YOU (AND NOT MICROSTRATEGY, INC. OR ANYONE ELSE WHO HAS BEEN INVOLVED WITH THE CREATION, PRODUCTION, OR DISTRIBUTION OF THE SOFTWARE OR DOCUMENTATION) ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICE, REPAIR, OR CORRECTION. SOME STATES DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.

In no event will MicroStrategy, Inc. or any other person involved with the creation, production, or distribution of the Software be liable to you on account of any claim for damage, including any lost profits, lost savings, or other special, incidental, consequential, or exemplary damages, including but not limited to any damages assessed against or paid by you to any third party, arising from the use, inability to use, quality, or performance of such Software and Documentation, even if MicroStrategy, Inc. or any such other person or entity has been advised of the possibility of such damages, or for the claim by any other party. In addition, MicroStrategy, Inc. or any other person involved in the creation, production, or distribution of the Software shall not be liable for any claim by you or any other party for damages arising from the use, inability to use, quality, or performance of such Software and Documentation, based upon principles of contract warranty, negligence, strict liability for the negligence of indemnity or contribution, the failure of any remedy to achieve its intended purpose or otherwise. The entire liability of MicroStrategy, Inc. and your exclusive remedy shall not exceed, at the option of MicroStrategy, Inc., either a full refund of the price paid, or replacement of the Software. No oral or written information given out expands the liability of MicroStrategy, Inc. beyond that specified in the above limitation of liability. Some states do not allow the limitation or exclusion of liability for incidental or consequential damages, so the above limitation may not apply to you.

The information contained in this manual (the Documentation) and the Software are copyrighted and all rights are reserved by MicroStrategy, Inc. MicroStrategy, Inc. reserves the right to make periodic modifications to the Software or the Documentation without obligation to notify any person or entity of such revision. Copying, duplicating, selling, or otherwise distributing any part of the Software or Documentation without prior written consent of an authorized representative of MicroStrategy, Inc. are prohibited. U.S. Government Restricted Rights. It is acknowledged that the Software and Documentation were developed at private expense, that no public part is known, and that the Software and Documentation are Commercial Computer Software provided with RESTRICTED RIGHTS under Federal Acquisition Regulations and agency supplements to them. Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFAR 252.227-7013 et seq. or subparagraphs (c)(1) and (2) of the Commercial Computer Software—Restricted Rights at FAR 52.227-19, as applicable.

Contractor is MicroStrategy, Inc., 1850 Towers Crescent Plaza, Vienna, VA 22182. Rights are reserved under copyright laws of the United States with respect to unused portions of the Software.


All other company and product names may be trademarks of the respective companies with which they are associated. Specifications subject to change without notice. MicroStrategy is not responsible for errors or omissions. MicroStrategy makes no warranties or commitments concerning the availability of future products or versions that may be planned or under development.

Patent Information

This product is patented. One or more of the following patents may apply to the product sold herein: U.S. Patent Nos. 6,154,766, 6,173,310, 6,260,050, 6,263,051, 6,269,393, 6,279,033, 6,567,796, 6,587,547, 6,606,596, 6,658,093, 6,658,432, 6,692,195, 6,671,715, 6,691,100, 6,694,316, 6,697,808, 6,704,723, 6,741,980, 6,765,997, 6,768,788, 6,772,137, 6,788,768, 6,789,987, 6,801,910, 6,820,073, 6,836,537, 6,850,603, 6,859,798, 6,873,693, 6,885,734, 6,940,953, 6,964,012, 6,977,992, 6,996,568, 6,996,569, 7,003,512, 7,010,518, 7,016,480, 7,020,251, 7,035,165, 7,082,422, 7,113,993, 7,127,403, 7,174,349, 7,181,417, 7,194,457, 7,197,461, 7,228,303, 7,260,577, 7,266,181, 7,272,212, 7,302,639, 7,324,942, 7,330,847, 7,340,040, 7,356,758, 7,356,840, 7,415,438, 7,428,302, 7,430,562, 7,440,898, 7,440,896, 7,486,780, 7,509,671, 7,516,181, 7,559,048, 7,574,376, 7,617,201, 7,617,201, 7,725,811, 7,801,967, 7,836,178, 7,861,161, 7,861,253, 7,881,443, 7,925,616, and 7,945,584. Other patent applications are pending.

Various MicroStrategy products contain the technology of third parties. This product may contain one or more of the following copyrighted technologies:

Graph Generation Engine Copyright © 1998-2011. Three D Graphics, Inc. All rights reserved.
Actuate® Formula One. Copyright © 1993-2011 Actuate Corporation. All rights reserved.
XML parser Copyright © 2003-2011 Microsoft Corporation. All rights reserved.
Xalan XSLT processor. Copyright © 1999-2011. The Apache Software Foundation. All rights reserved.
Xerces XML parser. Copyright © 1999-2011. The Apache Software Foundation. All rights reserved.
FOP XSL formatting objects. Copyright © 2004-2011. The Apache Software Foundation. All rights reserved.

Portions of Intelligence Server memory management Copyright © 1991-2011 Compuware Corporation. All rights reserved.

ASHHTTPRequest library. Copyright © 2007-2011, All-Seeing Interactive. All rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)
CONTENTS

Description of Guide ................................................................. xv
About this book ............................................................................ xvi
  Additional formats ................................................................. xvi
  How to find business scenarios and examples ....................... xvii
  What’s new in this guide .................................................... xviii
  Prerequisites ........................................................................ xviii
  Who should use this guide .................................................. xviii
Resources.................................................................................. xviii
  Documentation .................................................................... xviii
  Education ........................................................................... xxvii
  Consulting ......................................................................... xxvii
  International support ......................................................... xxvii
  Technical Support ............................................................ xxviii
Feedback ................................................................................. xxxiii

1. Planning Your Installation

  Introduction ............................................................................. 1
  Upgrade considerations ...................................................... 2
  MicroStrategy components ................................................ 2
    MicroStrategy Desktop products ...................................... 3
    MicroStrategy Command Manager .................................. 4
    MicroStrategy Object Manager ....................................... 4
    MicroStrategy Enterprise Manager ................................ 5
    MicroStrategy Intelligence Server ................................... 5
    MicroStrategy Web and Web Universal ......................... 8
    MicroStrategy Mobile ................................................... 12
    MicroStrategy SDK ....................................................... 13
    MicroStrategy Integrity Manager .................................... 15
## Contents

MicroStrategy Office ............................................................... 15  
MicroStrategy Narrowcast Server ........................................... 17  
MicroStrategy Analytics Modules ........................................... 18  
MicroStrategy Tutorial Reporting ........................................... 18  
MicroStrategy Health Center .................................................. 19  

Installation prerequisites .......................................................... 19  
Recommended installation location and example deployments ....................................................... 20  
Hardware requirements and recommendations ........................................................... 23  
Software requirements and recommendations ......................................................... 34  

Installation considerations ........................................................... 71  
System sizing guidelines .......................................................... 71  
International support .............................................................. 78  
MicroStrategy Web and Intelligence Server compatibility .......... 79  
Certified ODBC drivers for MicroStrategy Intelligence Server .................................................... 79  
Recommended system settings for UNIX and Linux ......... 82  

Methods of installation ................................................................ 84  
Graphical user interface ......................................................... 84  
Command line ........................................................................ 85  
Silent installation ................................................................... 85  

Licensing information ................................................................... 86  
Types of licenses ................................................................... 86  

Installation and configuration checklists ........................................... 87  
Installing and configuring MicroStrategy on Windows ........... 87  
Installing and configuring MicroStrategy on UNIX and Linux ........................................................................ 89  

### 2. Installing MicroStrategy on Windows

**Introduction** ............................................................................. 91  
Installation procedure ............................................................. 92  
Installing with the MicroStrategy Installation Wizard ........... 93  
Installation verification .............................................................. 108  
Installing with a guided MicroStrategy introduction ............ 109  
Configuring your MicroStrategy installation ......................... 109  

### 3. Installing MicroStrategy on UNIX and Linux

**Introduction** ............................................................................. 111  
Installation procedures on UNIX and Linux ......................... 112  
Different methods of installation ............................................. 113  
Installing with the MicroStrategy Installation Wizard ........... 114  
Unique post-installation configurations ................................. 125
### 4. Activating Your Installation

**Introduction** ................................................................. 131
Request an Activation Code ...................................................... 132
Activate your installation ........................................................... 134
Configuring your MicroStrategy installation ................................ 135
Server Activation FAQ ............................................................. 135

### 5. Configuring and Connecting Intelligence Server

**Introduction** ........................................................................... 143
Communicating with databases: ODBC and DSNs .......................... 144
   Setting up ODBC ................................................................. 146
   Defining DSNs ..................................................................... 151
   Testing ODBC connectivity .................................................. 155
Initial MicroStrategy configuration .............................................. 157
   Configuration Wizard prerequisites ...................................... 158
   Configuring MicroStrategy software .................................... 159
   Creating metadata, History List, and statistics
     repositories .......................................................................... 164
   Setting up MicroStrategy Intelligence Server ....................... 173
   Creating project sources ..................................................... 178
   Configuring MicroStrategy with a response file ..................... 183
Connecting to a data source ...................................................... 192
   Specifying warehouse connection information ..................... 193
Creating a project ...................................................................... 208
Configuring your MicroStrategy installation ............................... 208

### 6. Deploying MicroStrategy Web and Web Universal

**Introduction** ........................................................................... 209
Deploying with IIS (Windows) .................................................... 210
   Deploying MicroStrategy Web ............................................. 211
   Deploying Mobile Server ..................................................... 212
General steps to deploy MicroStrategy Web Universal .................. 214
   Directory structure after deploying the WAR file ................. 215
Deploying with WebLogic and Apache (Solaris) ......................... 216
   WebLogic paths and folder locations ................................... 216
   Preconfiguration information .............................................. 217
   Deploying MicroStrategy Web Universal ......................... 220
Re-deploy the application .................................................... 228
Performance-based setup information ............................ 229
Deploying with WebSphere and IBM HTTP Server (AIX) ........ 234
Preconfiguration information ........................................ 234
Deploying MicroStrategy Web Universal ....................... 236
Performance-based setup information ............................ 244
Deploying with Sun Java System (Solaris) ....................... 246
Preconfiguration information ........................................ 247
Deploying MicroStrategy Web Universal ....................... 249
Performance-based setup information ............................ 256
Deploying with Tomcat (Windows) ................................... 257
Preconfiguration information ........................................ 257
Deploying MicroStrategy Web Universal ....................... 261
Deploying with Tomcat (Linux) ....................................... 264
Preconfiguration information ........................................ 264
Deploying MicroStrategy Web Universal ....................... 268
Deploying with SAP NetWeaver (Windows) ..................... 270
Preconfiguration information ........................................ 271
Deploying MicroStrategy Web Universal ....................... 272
Deploying with Oracle 10g (Windows) ............................ 275
Preconfiguration information ........................................ 275
Deploying MicroStrategy Web Universal ....................... 277
Deploying with JBoss (Windows) ...................................... 280
Preconfiguration information ........................................ 281
Deploying MicroStrategy Web Universal ....................... 283
Administering your MicroStrategy Web deployment .......... 286
Enabling users to install MicroStrategy Office from Web ... 287
Configuring your MicroStrategy installation ................. 288

7. Deploying MicroStrategy Web Services

Introduction .............................................................................. 289
Deploying with IIS (Windows) .............................................. 290
General steps to deploy MicroStrategy Web Services J2EE .. 290
Deploying with WebLogic ...................................................... 291
WebLogic paths and folder locations .............................. 292
Preconfiguration information ........................................ 293
Deploying MicroStrategy Web Services J2EE ................. 294
Redeploying the application ............................................ 300
Performance-based setup information ............................ 300
Deploying with WebSphere and IBM HTTP Server ........... 301
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preconfiguration information</td>
<td>302</td>
</tr>
<tr>
<td>Deploying MicroStrategy Web Services J2EE</td>
<td>303</td>
</tr>
<tr>
<td>Performance-based setup information</td>
<td>310</td>
</tr>
<tr>
<td>Deploying with Sun Java System</td>
<td>311</td>
</tr>
<tr>
<td>Preconfiguration information</td>
<td>312</td>
</tr>
<tr>
<td>Deploying MicroStrategy Web Services J2EE</td>
<td>313</td>
</tr>
<tr>
<td>Performance-based setup information</td>
<td>319</td>
</tr>
<tr>
<td>Deploying with Tomcat</td>
<td>320</td>
</tr>
<tr>
<td>Preconfiguration information</td>
<td>321</td>
</tr>
<tr>
<td>Deploying MicroStrategy Web Services J2EE</td>
<td>321</td>
</tr>
<tr>
<td>Deploying with SAP</td>
<td>324</td>
</tr>
<tr>
<td>Preconfiguration information</td>
<td>324</td>
</tr>
<tr>
<td>Deploying MicroStrategy Web Services J2EE</td>
<td>325</td>
</tr>
<tr>
<td>Deploying with Oracle 10g</td>
<td>327</td>
</tr>
<tr>
<td>Preconfiguration information</td>
<td>327</td>
</tr>
<tr>
<td>Deploying MicroStrategy Web Services J2EE</td>
<td>328</td>
</tr>
<tr>
<td>Completing your MicroStrategy Web Services J2EE deployment</td>
<td>331</td>
</tr>
<tr>
<td>Validating your MicroStrategy Web Services J2EE deployment</td>
<td>331</td>
</tr>
<tr>
<td>Editing the project sources file</td>
<td>332</td>
</tr>
<tr>
<td>Verify Web Services connectivity with Intelligence Server</td>
<td>333</td>
</tr>
<tr>
<td>Administering your MicroStrategy Web Services deployment</td>
<td>333</td>
</tr>
<tr>
<td>Configuring your MicroStrategy installation</td>
<td>334</td>
</tr>
<tr>
<td>8. Setting Up Documents and HTML Documents</td>
<td>335</td>
</tr>
<tr>
<td>Introduction</td>
<td>335</td>
</tr>
<tr>
<td>Prerequisites</td>
<td>336</td>
</tr>
<tr>
<td>Executing documents and HTML documents in UNIX/Linux</td>
<td>336</td>
</tr>
<tr>
<td>Setup for creating and executing HTML documents</td>
<td>337</td>
</tr>
<tr>
<td>Setup for executing existing HTML documents</td>
<td>341</td>
</tr>
<tr>
<td>Setup for executing Report Services documents</td>
<td>342</td>
</tr>
<tr>
<td>Configuring your MicroStrategy installation</td>
<td>343</td>
</tr>
<tr>
<td>9. Automated Installation on Windows</td>
<td>345</td>
</tr>
<tr>
<td>Introduction</td>
<td>345</td>
</tr>
<tr>
<td>Installation log file</td>
<td>346</td>
</tr>
<tr>
<td>Methods of installation</td>
<td>347</td>
</tr>
<tr>
<td>Installing and configuring with a response.ini file</td>
<td>347</td>
</tr>
<tr>
<td>Configuring a response.ini file to install MicroStrategy</td>
<td>348</td>
</tr>
</tbody>
</table>

Introduction .............................................................................. 427
Creating a DSN for a data source .............................................. 428
Testing and examining ODBC connectivity ............................... 429
Configuring MicroStrategy with a response.ini file ..................... 430
Creating a response.ini file .................................................. 431
Using the response.ini file to configure MicroStrategy .......... 448
Parameters and options in the response.ini file ....................... 448
Configuring and controlling Intelligence Server ......................... 449
Starting, configuring, and monitoring Intelligence Server with mstrsvr .................................................. 449
Configuring the default server instance with mstrsvr-configure ......................................................................... 450
Creating and configuring Intelligence Server instances with mstrctl ................................................................. 451
Supporting reserved words and characters ............................... 453
Configuring your MicroStrategy installation ............................... 454

13. Adding or Removing MicroStrategy Components

Introduction .............................................................................. 455
Adding or removing MicroStrategy components on Windows ... 456
Re-installing MicroStrategy components on Windows.............. 457
Re-installing MicroStrategy Office ........................................ 458
Uninstalling MicroStrategy components on Windows.............. 459
Uninstalling MicroStrategy Office ......................................... 460
Uninstalling MicroStrategy components on UNIX and Linux ..... 461

A. Connecting to Databases: ODBC and DSNs

Introduction .............................................................................. 463
Creating DSNs for specific ODBC drivers ................................. 463
MicroStrategy ODBC Driver for DB2 Wire Protocol for UDB for Windows ................................................................. 467
MicroStrategy ODBC Driver for DB2 UDB for UNIX/Linux ......................................................................................... 467
MicroStrategy ODBC Driver for DB2 Wire Protocol for UDB iSeries/DB2 for i for Windows and UNIX/Linux ................. 468
MicroStrategy ODBC Driver for DB2 z/OS for Windows and UNIX/Linux ................................................................. 469
MicroStrategy ODBC Driver for Greenplum Wire Protocol for Windows and UNIX/Linux ............................................ 469
MicroStrategy ODBC Driver for Informix Wire Protocol for Windows and UNIX/Linux ........................................... 470
MicroStrategy ODBC Driver for Informix 8 for Windows ......... 470
Graph and document support of non-Western European fonts ............................................................... 530
Server port number errors ................................................................. 531
  I forgot the server port number ................................................... 531
  Port number is in use ............................................................... 531
DSN connection errors ........................................................................ 532
  Testing the DSN connection failed in DSN Creator ............... 532
Permission errors ........................................................................... 532
  Missing JVM file .................................................................... 532

C. Configuring a web.xml File to Use Absolute Paths

Introduction .......................................................................................... 535
Example of modified web.xml file .................................................. 535

Glossary ................................................................................................. 545

Index ....................................................................................................... 551
Description of Guide

The MicroStrategy Installation and Configuration Guide provides comprehensive information on how to install and configure MicroStrategy software, as well as basic maintenance guidelines. This guide gets you started using the UNIX, Linux, and Windows versions of the MicroStrategy platform.

For UNIX and Linux installations, this guide assumes a basic understanding of how to use UNIX and Linux either from a UNIX/Linux server or by using a viewer on a PC.

For tasks that require advanced system administration commands, this document assumes you are either familiar with those commands or can contact your system administrator.

The main chapters of this guide follow the recommended progression of steps to install and configure MicroStrategy. Refer to Chapter 1, Planning Your Installation, for important installation prerequisites before you begin installing MicroStrategy.

For details on how to progress through the remaining chapters of this guide, see the section Installation and configuration checklists, page 87 of Chapter 1, Planning Your Installation.

The appendixes contain the following additional reference information, which you may require depending on your specific needs:

- Appendix A, Connecting to Databases: ODBC and DSNs, provides details and steps to configure ODBC and DSNs for your data warehouse connections.
• *Appendix B, Troubleshooting*, provides various troubleshooting steps and techniques to take in certain installation and configuration scenarios.

• *Appendix C, Configuring a web.xml File to Use Absolute Paths*, provides a sample web.xml file configured to use absolute paths. This modification is necessary for MicroStrategy Web Universal to deploy correctly in certain scenarios.

**About this book**

The chapters in this book provide conceptual information about:

• All MicroStrategy components
• Installation and configuration procedures
• System tuning considerations
• Troubleshooting and maintenance guidelines

Each chapter begins with a brief overview of the chapter’s content.

Dates in the MicroStrategy Tutorial project are updated to reflect the current year. The sample documents and images in this guide, as well as the procedures, were created with dates that may no longer be available in the Tutorial project. Replace them with the first year of data in your Tutorial project.

**Additional formats**

This book is also available as an electronic publication in the Apple iBookstore, and can be read on an iPhone or iPad with the iBooks app installed. To download this book, search for the book’s title in the iBookstore search bar, or scan the QR code below using your device's camera.
How to find business scenarios and examples

Within this guide, many of the concepts discussed are accompanied by business scenarios or other descriptive examples, where applicable. Many of the examples use the MicroStrategy Tutorial, which is MicroStrategy’s sample warehouse, metadata, and project. Information about the MicroStrategy Tutorial may be found in the MicroStrategy Basic Reporting Guide.

Other examples in this book use the Analytics Modules, which include a set of precreated sample reports, each from a different business area. Sample reports present data for analysis in such business areas as financial reporting, human resources, and customer analysis.

Detailed examples of advanced reporting functionality can be found in the MicroStrategy Advanced Reporting Guide.

What’s new in this guide

Each release of the Installation and Configuration Guide is updated to reflect the installation requirements of the associated version of the MicroStrategy Product Suite. The sections below document the additional improvements to the documentation for recent releases.

MicroStrategy 9.2.1

- Information has been provided on a new product, MicroStrategy Transaction Services (see MicroStrategy Transaction Services, page 8).

MicroStrategy 9.2.0

- The steps to perform an OEM silent installation have been updated (see OEM silent installations, page 418).
Prerequisites

Before working with this document, you should be familiar with

- The nature and structure of the data to use for your business intelligence application
- Your system’s configuration, including details such as hardware configuration, installed applications, available memory, and so on
- If installing on UNIX or Linux, the basics of the UNIX or Linux operating system

Who should use this guide

This document is designed for system administrators who install, configure, and maintain MicroStrategy software on the UNIX, Linux, or Windows operating systems.

This document discusses how to perform automated and silent installations. Automated and silent installations require advanced techniques such as creating and running response.ini files. Therefore, automated and silent installations should be handled by system administrators with full knowledge of the environment and the desired MicroStrategy installation.

Resources

Documentation

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

- **Manuals**: In general, 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 xxiv*. Most of these manuals are also available printed in a bound, soft cover format. To purchase printed manuals, contact your MicroStrategy account executive or email documentationfeedback@microstrategy.com.

- **Help:** In general, MicroStrategy help provides:
  - Detailed steps to perform procedures
  - Descriptions of each option on every software screen

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. For the most up-to-date translations, refer to the MicroStrategy Knowledge Base.

**MicroStrategy overview and evaluation**

- **Introduction to MicroStrategy: Evaluation Guide**

  Instructions for installing, configuring, and using the MicroStrategy Evaluation Edition of the software. This guide also includes a detailed, step-by-step evaluation process of MicroStrategy features, where you perform reporting with the MicroStrategy Tutorial project and its sample business data.


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

- **Evaluate MicroStrategy for Linux Guide: In a Windows or Linux Environment with the MicroStrategy Evaluation Edition Virtual Appliance**

  Evaluate MicroStrategy for Linux, in a Microsoft Windows or Linux environment, with the MicroStrategy Evaluation Edition Virtual Appliance. This guide provides all details to download, activate, and evaluate MicroStrategy software running in a Linux environment.

- **MicroStrategy Reporting Suite: Quick Start Guide**

  Evaluate MicroStrategy as a departmental solution. Provides detailed information to download, install, configure, and use the MicroStrategy Reporting Suite.
• **MicroStrategy Mobile Suite: Quick Start Guide**

Evaluate MicroStrategy Mobile as a departmental solution. Provides detailed information to download, install, configure, and use the MicroStrategy Mobile Suite.

**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, as well as basic maintenance guidelines.

• **MicroStrategy Upgrade Guide**

Instructions to upgrade existing MicroStrategy products.

• **MicroStrategy Project Design Guide**

Information to create and modify MicroStrategy projects, and understand facts, attributes, hierarchies, transformations, advanced schemas, and project optimization.

• **MicroStrategy Basic Reporting Guide**

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

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

Instructions for advanced topics in the MicroStrategy system, building on information in the Basic Reporting Guide. Topics include reports, Freeform SQL reports, Query Builder reports, filters, metrics, Data Mining Services, custom groups, consolidations, and prompts.


Instructions for a business analyst to execute and analyze a document in MicroStrategy Desktop and MicroStrategy Web, building on basic concepts about projects and reports presented in the MicroStrategy Basic Reporting Guide.
• MicroStrategy Report Services Document Creation Guide: Creating Boardroom Quality Documents


• MicroStrategy OLAP Services Guide

Information on MicroStrategy OLAP Services, which is an extension of MicroStrategy Intelligence Server. OLAP Services features include Intelligent Cubes, derived metrics, derived elements, dynamic aggregation, view filters, and dynamic sourcing.

• MicroStrategy Office User Guide

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

• MicroStrategy Mobile Analysis Guide: Analyzing Data with MicroStrategy Mobile

Information and instructions for using 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 instructions for a designer working in MicroStrategy Desktop 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

Concepts and high-level 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

Information and instructions for MicroStrategy 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.

**Manuals for Analytics Modules**

• **Analytics Modules Installation and Porting Guide**

• **Customer Analysis Module Reference**

• **Sales Force Analysis Module Reference**

• **Financial Reporting Analysis Module Reference**

• **Sales and Distribution Analysis Module Reference**

• **Human Resources Analysis Module Reference**

**Manuals for Narrowcast Services products**

• **MicroStrategy Narrowcast Server Getting Started Guide**  
  Instructions to work with the tutorial to learn Narrowcast Server interfaces and features.

• **MicroStrategy Narrowcast Server Installation and Configuration Guide**  
  Information to install and configure Narrowcast Server.

• **MicroStrategy Narrowcast Server Application Designer Guide**  
  Fundamentals of designing Narrowcast Server applications.

• **MicroStrategy Narrowcast Server System Administrator Guide**  
  Concepts and high-level steps to implement, maintain, tune, and troubleshoot Narrowcast Server.

• **MicroStrategy Narrowcast Server Upgrade Guide**  
  Instructions to upgrade an existing Narrowcast Server.
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 sold as part of the MicroStrategy SDK.

• **Narrowcast Server SDK Guide**

  Instructions to customize Narrowcast Server functionality, integrate Narrowcast Server with other systems, and embed Narrowcast Server functionality within other applications. Documents the Narrowcast Server Delivery Engine and Subscription Portal APIs, and the Narrowcast Server SPI.

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 can be accessed using the MicroStrategy Product Manuals page, as described in *Accessing manuals and other documentation sources, page xxiv*.

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 ESRI mapping services.

  This resource can be accessed using the MicroStrategy Product Manuals page, as described in *Accessing manuals and other documentation sources, page xxiv*. 
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 your MicroStrategy disk or the machine where MicroStrategy was installed.

Adobe Acrobat Reader is required to view these manuals. If you do not have Acrobat 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 installed manuals and other documentation sources on Windows, page xxv
To access installed manuals and other documentation sources on UNIX and Linux, page xxv

To access installed manuals and other documentation sources on Windows

1 From the Windows Start menu, choose Programs (or All Programs), MicroStrategy, 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.

3 If you click the link for the Narrowcast Services SDK Guide, a File Download dialog box opens. This documentation resource must be downloaded. Select Open this file from its current location, and click OK.

If bookmarks are not visible on the left side of an Acrobat (PDF) manual, from the View menu click Bookmarks and Page. This step varies slightly depending on your version of Adobe Acrobat Reader.

To access installed manuals and other documentation sources 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 Documentation 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.

5 If you click the link for the Narrowcast Services SDK Guide, a File Download dialog box opens. This documentation resource must be
downloaded. Select Open this file from its current location, and click OK.

If bookmarks are not visible on the left side of an Acrobat (PDF) manual, from the View menu click Bookmarks and Page. This step varies slightly depending on your version of Adobe Acrobat 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.

<table>
<thead>
<tr>
<th>Type</th>
<th>Indicates</th>
</tr>
</thead>
</table>
| **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  
                • Text to be entered by the user  
                Example: Click Select Warehouse.  
                Example: Type cmdmgr -f scriptfile.scp and press Enter. |
| **italic**   | • New terms defined within the text and in the glossary  
                • Names of other product manuals  
                • When part of a command syntax, indicates variable information to be replaced by the user  
                Example: The aggregation level is the level of calculation for the metric.  
                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  
                Example: Sum(revenue)/number of months. |
| +            | 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. For a detailed description of education offerings and course curriculums, 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. For a detailed description of consulting offerings, visit http://www.microstrategy.com/Consulting.

International support

MicroStrategy supports several locales. Support for a locale typically includes native database and operating system support, support for date formats, numeric formats, currency symbols, and availability of translated interfaces and certain documentation.

MicroStrategy is certified in homogeneous configurations (where all the components lie in the same locale) in the following languages—English (US), French, German, Italian, Japanese, Korean, Portuguese (Brazilian), Spanish, Chinese (Simplified), Chinese (Traditional), Danish, and Swedish. A translated user interface is available in each of the above languages. For information on specific languages supported by individual MicroStrategy system components, see the MicroStrategy readme.

MicroStrategy also provides limited support for heterogeneous configurations (where some of the components may lie in different locales). Please contact MicroStrategy Technical Support for more details.

A translated user interface is available in each of the above languages.
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 If the resources listed in the steps above do not provide a solution, contact MicroStrategy Technical Support directly. To ensure the most productive relationship with MicroStrategy Technical Support, review the Policies and Procedures document in your language, posted at http://www.microstrategy.com/Support/Policies. Refer to the terms of your purchase agreement to determine the type of support available to you.

MicroStrategy Technical Support can be contacted by your company’s Support Liaison. A Support Liaison is a person whom your company has designated as a point-of-contact with MicroStrategy’s support personnel. All customer inquiries and case communications must come through these named individuals. Your company may designate two employees to serve as their Support Liaisons, and can request to change their Support Liaisons two times per year with prior written notice to MicroStrategy Technical Support.

It is recommended that you designate Support Liaisons who have MicroStrategy Administrator privileges. This can eliminate security conflicts and improve case resolution time. When troubleshooting and researching issues, MicroStrategy Technical Support personnel may make recommendations that require administrative privileges within MicroStrategy, or that assume that the designated Support Liaison has a security level that permits them to fully manipulate the MicroStrategy projects and has access to potentially sensitive project data such as security filter definitions.
Ensure issues are resolved quickly

Before logging a case with MicroStrategy Technical Support, the Support Liaison may follow the steps below to ensure that issues are resolved quickly:

1 Verify that the issue is with MicroStrategy software and not a third party software.

2 Verify that the system is using a currently supported version of MicroStrategy software by checking the Product Support Expiration Schedule at http://www.microstrategy.com/Support/Expiration.asp.

3 Attempt to reproduce the issue and determine whether it occurs consistently.

4 Minimize the complexity of the system or project object definition to isolate the cause.

5 Determine whether the issue occurs on a local machine or on multiple machines in the customer environment.

6 Discuss the issue with other users by posting a question about the issue on the MicroStrategy Customer Forum at https://resource.microstrategy.com/forum/.

The following table shows where, when, and how to contact MicroStrategy Technical Support. If your Support Liaison is unable to reach MicroStrategy Technical Support by phone during the hours of operation, they can leave a voicemail message, send email or fax, or log a case using the Online Support
Interface. The individual Technical Support Centers are closed on certain public holidays.

<table>
<thead>
<tr>
<th>Region</th>
<th>Email</th>
<th>Web</th>
<th>Fax</th>
<th>Phone</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>North America</td>
<td><a href="mailto:support@microstrategy.com">support@microstrategy.com</a></td>
<td><a href="https://resource.microstrategy.com/support">https://resource.microstrategy.com/support</a></td>
<td>(703) 842–8709</td>
<td>(703) 848–8700</td>
<td>9:00 A.M.–7:00 P.M. Eastern Time, Monday–Friday except holidays</td>
</tr>
<tr>
<td>EMEA: Europe</td>
<td><a href="mailto:eurosupp@microstrategy.com">eurosupp@microstrategy.com</a></td>
<td><a href="https://resource.microstrategy.com/support">https://resource.microstrategy.com/support</a></td>
<td>+44 (0) 208 711 2525</td>
<td></td>
<td>The European Technical Support Centre is closed on national public holidays in each country.</td>
</tr>
<tr>
<td>The Middle East</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Phone:</td>
</tr>
<tr>
<td>Africa</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Belgium: + 32 2792 0436</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• France: +33 17 099 4737</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Germany: +49 22 16501 0609</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Ireland: +353 1436 0916</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Italy: +39 023626 9668</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Poland: +48 22 321 8680</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Scandinavia &amp; Finland: +46 8505 20421</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Spain: +34 91788 9852</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• The Netherlands: +31 20 794 8425</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• UK: +44 (0) 208 080 2182</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• International distributors: +44 (0) 208 080 2183</td>
</tr>
<tr>
<td>Asia Pacific</td>
<td><a href="mailto:apsupport@microstrategy.com">apsupport@microstrategy.com</a></td>
<td><a href="https://resource.microstrategy.com/support">https://resource.microstrategy.com/support</a></td>
<td></td>
<td></td>
<td>Japan and Korea: 9:00 A.M.–6:00 P.M. JST (Tokyo), Monday–Friday except holidays</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Asia Pacific (except Japan and Korea): 7 A.M.–6 P.M. (Singapore) Monday–Friday except holidays</td>
</tr>
<tr>
<td>Latin America</td>
<td><a href="mailto:latamsupport@microstrategy.com">latamsupport@microstrategy.com</a></td>
<td><a href="https://resource.microstrategy.com/support">https://resource.microstrategy.com/support</a></td>
<td></td>
<td></td>
<td>Latin America (except Brazil): 9:00 A.M.–7:00 P.M. (Buenos Aires), Monday–Friday except holidays</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Brazil: 9 A.M. – 6 P.M. (São Paulo), Monday–Friday except holidays</td>
</tr>
</tbody>
</table>
Support Liaisons should contact the Technical Support Center from which they obtained their MicroStrategy software licenses or the Technical Support Center to which they have been designated.

**Required information when calling**

When contacting MicroStrategy Technical Support, please provide the following information:

- **Personal information:**
  - Name (first and last)
  - Company and customer site (if different from company)
  - Contact information (phone and fax numbers, e-mail addresses)

- **Case details:**
  - Configuration information, including MicroStrategy software product(s) and versions
  - Full description of the case including symptoms, error messages(s), and steps taken to troubleshoot the case thus far

- **Business/system impact**

If this is the Support Liaison’s first call, they should also be prepared to provide the following:

- Street address
- Phone number
- Fax number
- Email address

To help the Technical Support representative resolve the problem promptly and effectively, be prepared to provide the following additional information:

- Case number: Please keep a record of the number assigned to each case logged with MicroStrategy Technical Support, and be ready to provide it when inquiring about an existing case

- Software version and product registration numbers of the MicroStrategy software products you are using
• Case description:
  ▪ What causes the condition to occur?
  ▪ Does the condition occur sporadically or each time a certain action is performed?
  ▪ Does the condition occur on all machines or just on one?
  ▪ When did the condition first occur?
  ▪ What events took place immediately prior to the first occurrence of the condition (for example, a major database load, a database move, or a software upgrade)?
  ▪ If there was an error message, what was its exact wording?
  ▪ What steps have you taken to isolate and resolve the issue? What were the results?

• System configuration (the information needed depends on the nature of the problem; not all items listed below may be necessary):
  ▪ Computer hardware specifications (processor speed, RAM, disk space, and so on)
  ▪ Network protocol used
  ▪ ODBC driver manufacturer and version
  ▪ Database gateway software version
  ▪ (For MicroStrategy Web-related problems) browser manufacturer and version
  ▪ (For MicroStrategy Web-related problems) Web server manufacturer and version

If the issue requires additional investigation or testing, the Support Liaison and the MicroStrategy Technical Support representative should agree on certain action items to be performed. The Support Liaison should perform any agreed-upon actions before contacting MicroStrategy Technical Support again regarding the issue. If the Technical Support representative is responsible for an action item, the Support Liaison may call MicroStrategy Technical Support at any time to inquire about the status of the issue.
Feedback

Please send any comments or suggestions about user documentation for MicroStrategy products to:

documentationfeedback@microstrategy.com

Send suggestions for product enhancements to:

support@microstrategy.com

When you provide feedback to us, please include the name and version of the products you are currently using. Your feedback is important to us as we prepare for future releases.
Introduction

MicroStrategy business intelligence tools help organizations to monitor, report, and analyze all of their enterprise data. MicroStrategy helps you make decisions based upon the data within your organization’s enterprise data warehouse.

This chapter provides an overview of the different MicroStrategy components and products so you can decide what you need to install. It also provides details on supported functionality and describes important installation prerequisites that should be considered before you start installing MicroStrategy products.

The MicroStrategy products that you can install depend on your MicroStrategy license. Contact your MicroStrategy account executive with MicroStrategy licensing questions.

This chapter provides the following information:

- Upgrade considerations, page 2
- MicroStrategy components, page 2
- Installation prerequisites, page 19
Upgrade considerations

If you have MicroStrategy products installed and want to upgrade them, refer to the MicroStrategy Upgrade Guide. The upgrade procedures on Windows, UNIX, and Linux are covered in this guide. It is recommended that you read this guide before deciding to upgrade an existing metadata.

MicroStrategy components

MicroStrategy has a range of products that you can install on different operating systems. Depending on the type of setup you have, you can install various combinations of MicroStrategy components. The components described in this section offer a complete set of tools for creating, deploying, supporting, and maintaining the business intelligence applications. This section briefly describes the most common setups and the components required for each.

If you are familiar with MicroStrategy components and subcomponents, you can skip this section and go to Installation prerequisites, page 19.

This section discusses MicroStrategy components and their subcomponents in relation to how the components are grouped together during the installation routine. These groups are meant to help guide you in selecting MicroStrategy products during your installation. The groups do NOT represent MicroStrategy product license requirements. For information on license requirements for MicroStrategy products, contact your MicroStrategy account executive.
MicroStrategy Desktop products

MicroStrategy Desktop provides analytical features designed to facilitate and perform the deployment of reports. It governs application objects such as reports, filters, and metrics.

Desktop also enables you to create application objects. The application objects are built on the *schema objects* that are created in MicroStrategy Architect. These application objects are used to analyze and provide insight into relevant data. The following sections provide a brief description of the subcomponents for these products.

MicroStrategy Desktop products subcomponents

For details about the specific features that the different Desktop versions include, review the feature comparison chart at the following website:

http://www.microstrategy.com/Software/Products/User_Interfaces/Desktop/version_features.asp

- MicroStrategy Desktop Analyst is a simplified version of MicroStrategy Desktop, providing the basic interactive reporting functionality required by managers.

- MicroStrategy Desktop Designer is a full-featured version for power analysts and application developers. With a full range of analytical functionality, a rich library of functions, and intelligent workflow, Desktop Designer is well suited for power users.

- MicroStrategy Architect provides project designer functionality such as attribute, fact, hierarchy, and project creation and modification. Architect contains the following subcomponents:
  - *MicroStrategy Architect, page 4*, which is a new project design tool.
  - MicroStrategy Project Builder allows quick creation of simple projects. These projects can be used for easy creation of prototypes for proof-of-concept purposes.
  - MicroStrategy Function Plug-in Wizard is an add-in to the Microsoft Visual C++ compiler, which comes with a standard MicroStrategy installation. It allows you to create a C++ project, with which you can implement your own custom MicroStrategy function plug-in. The option to install this component is enabled only if Microsoft Visual C++ 8.0 is present on the system where the installation is being performed.
• MicroStrategy Server Administrator is a MicroStrategy Intelligence Server administrative console that provides functionality such as system monitoring, cache management, and user and group management.

When installing MicroStrategy Desktop, your license key must be licensed for MicroStrategy Intelligence Server to install and access MicroStrategy Server Administrator.

For information on various options present in Desktop to create and run reports, refer to the MicroStrategy Basic Reporting Guide. After you are familiar with basic Desktop concepts, refer to the MicroStrategy Advanced Reporting Guide for information on advanced Desktop functionality.

**MicroStrategy Architect**

MicroStrategy 9.0 introduces a new project design tool known as Architect. Architect allows you to define all the required components of your project from a centralized interface. Architect also provides a visual representation of your project as you create it, which helps to provide an intuitive workflow.

For information on using Architect to design a project in MicroStrategy, see the Project Design Guide.

**MicroStrategy Command Manager**

MicroStrategy Command Manager is an application designed to simplify and automate administration tasks, such as add, delete, or update enterprise-level data associated with large numbers of users and user groups. Additionally, Command Manager allows you to manage various configuration settings within the MicroStrategy platform.

For information on using Command Manager, see the System Administration Guide.

**MicroStrategy Object Manager**

MicroStrategy Object Manager provides complete life cycle management capabilities for MicroStrategy environments. Using Object Manager, you can copy objects within a project or across related projects.
For information on using Object Manager, see the *System Administration Guide*.

**MicroStrategy Enterprise Manager**

MicroStrategy Enterprise Manager provides insights about governing and tuning all areas of your MicroStrategy environment. With Enterprise Manager, you can see a variety of Intelligence Server usage statistics. The statistics shown in predefined reports displayed by Enterprise Manager can help you make scheduling decisions, analyze bottlenecks, and tune performance.

If you install MicroStrategy on Windows XP, the option to install Enterprise Manager is not displayed.

For information on using Enterprise Manager, see the *System Administration Guide*.

**MicroStrategy Intelligence Server**

MicroStrategy Intelligence Server delivers world-class monitoring, reporting, and analysis on a single integrated platform, offering next generation BI capabilities for the full range of BI applications. MicroStrategy Intelligence Server is the architectural foundation of the MicroStrategy platform. It performs the following critical tasks for the MicroStrategy BI platform:

- Runs queries, performs calculations, and formats reports
- Significantly improves user-perceived query performance
- Efficiently manages thousands of end-user requests (jobs)
- Serves as a central point for the MicroStrategy metadata

Intelligence Server also provides a library of over 150 different sophisticated mathematical and statistical functions, which can be added to. See the *Functions Reference* for details about these functions.

All other products in the platform work in conjunction with Intelligence Server and benefit from its broad functionality.
MicroStrategy has different editions of Intelligence Server, which are:

- Standard edition
- Enterprise edition
- Universal edition

The Standard edition and Enterprise edition are Windows versions of Intelligence Server, whereas the Universal edition is platform-independent. The Universal edition can be installed on various operating systems such as Windows, UNIX, Linux, and so on. The Standard, Enterprise, and Universal editions of Intelligence Server share a common code base and run reports identically. However, there are a few functionality differences. The following table highlights the major differences between them.

<table>
<thead>
<tr>
<th>Functionality</th>
<th>Intelligence Server Universal Edition</th>
<th>Intelligence Server Standard and Enterprise Editions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows Authentication</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td>Custom Function Plug-in</td>
<td>Not supported</td>
<td>Supported</td>
</tr>
<tr>
<td>Intelligence Server SDK</td>
<td>MicroStrategy SDK applications can be run against Intelligence Server Universal Edition, but they need to be executed from a Windows machine</td>
<td>Supported</td>
</tr>
<tr>
<td>Microsoft Access</td>
<td>Supported, through the use of the MicroStrategy ODBC Driver for Sequelink (see MicroStrategy ODBC Driver for Sequelink, page 475)</td>
<td>Supported</td>
</tr>
<tr>
<td>Operating System Support</td>
<td>Supports Windows and UNIX/Linux operating systems for all applicable CPU architectures. For information on the supported CPU architectures, see:</td>
<td>Supports only Windows 32-bit operating systems.</td>
</tr>
<tr>
<td></td>
<td>- UNIX/Linux: Intelligence Server Universal software requirements on UNIX/Linux, page 52</td>
<td></td>
</tr>
</tbody>
</table>

**MicroStrategy Intelligence Server subcomponents**

The subcomponents of MicroStrategy Intelligence Server are as follows:

- Universal Edition, which is described in MicroStrategy Intelligence Server, page 5.
- **MicroStrategy OLAP Services, page 7.**
- **MicroStrategy Report Services, page 7.**
- **MicroStrategy Distribution Services, page 7.**
- **MicroStrategy Transaction Services, page 8.**
- **MultiSource Option, page 8.**

Clustering Option, which allows you to cluster a group of Intelligence Server machines (up to four Intelligence Server machines) to take advantage of the many benefits available in a clustered environment.

For information on clustering Intelligence Servers, see the System Administration Guide.

**MicroStrategy OLAP Services**

MicroStrategy OLAP Services uses the concept of Intelligent Cube, an in-memory version of a report that can be manipulated by the Analytical Engine. MicroStrategy Desktop, MicroStrategy Web, and MicroStrategy Office users can slice and dice data in reports within the Intelligent Cubes without having to re-execute SQL against the data warehouse.

For information on OLAP Services, see the OLAP Services Guide.

**MicroStrategy Report Services**

MicroStrategy Report Services is the enterprise reporting engine of the MicroStrategy business intelligence platform. A MicroStrategy Report Services document contains objects representing data coming from one or more reports, as well as positioning and formatting information. It is used to format data from multiple reports in a single display of presentation quality.


**MicroStrategy Distribution Services**

MicroStrategy Transaction Services

MicroStrategy Transaction Services is an Intelligence Server extension that lets you embed write-back functionality into reports and dashboards for the purposes of decision-making or initiating a transaction. These transactions can include one-click approvals and denials, notes for tracking and directing business activity, and write-back to data sources in real time.

Users of MicroStrategy Web, MicroStrategy Mobile for iPhone, and MicroStrategy Mobile for iPad can employ these transaction capabilities from reports, dashboards, and MicroStrategy Mobile applications.

MultiSource Option

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 purpose. All data sources included the MultiSource Option are integrated as part of the same relational schema for a project.

For information on using MultiSource Option, see the Project Design Guide.

MicroStrategy Web and Web Universal

MicroStrategy implements Web and Web Universal using the .NET and JAVA technologies. MicroStrategy Web is .NET (ASP.NET) and MicroStrategy Web Universal is Java (JSP) based. Hence, MicroStrategy Web can be deployed only on Windows environments, whereas MicroStrategy Web Universal can be deployed on UNIX, Linux, and Windows.

MicroStrategy Web provides users with a highly interactive environment and a low-maintenance interface for reporting and analysis. Using the MicroStrategy Web interface, users can access, analyze, and share corporate data through any Web browser on any operating system. MicroStrategy Web provides ad hoc querying, industry-leading analysis, quick deployment, and rapid customization, making it even easier for users to make informed business decisions.
In addition to the powerful functionality that MicroStrategy Web offers, MicroStrategy Web Universal provides the added benefit of working with:

- Operating systems such as Oracle Solaris, IBM AIX, and HP-UX
- Application servers such as Oracle WebLogic and IBM WebSphere
- Web servers and Web browsers

For information on how to deploy MicroStrategy Web and Web Universal, see *Chapter 6, Deploying MicroStrategy Web and Web Universal*.

For detailed information about using the MicroStrategy Web reporting environment, refer to the online help provided on the MicroStrategy Web interface.

For information about configuring and tuning MicroStrategy Web, refer to the *MicroStrategy System Administration Guide*.

**MicroStrategy Web and Web Universal versions**

MicroStrategy Web and Web Universal are available in the following versions:

- **Web Reporter**: Business users are able to view all types of reports and scorecards and also personalize reports, print, drill, sort, export, choose between grid or graph format, and schedule or immediately send reports via email or to a file server or a printer.

- **Web Analyst**: This version provides all the functionality of Web Reporter plus the ability to drill anywhere, edit totals, pivot reports, add or remove fields from reports, create derived metrics, and create reports or ad hoc queries.

- **Web Professional**: This full-featured version provides all the functionality of Web Analyst plus the ability to design scorecards, dashboards, and operational reports in design mode or WYSIWYG view mode. Web Professional users have advanced formatting capabilities as well as the ability to perform calculations across multiple data sources.

   Intelligence Server license options provide access to certain services in MicroStrategy Web and Web Universal versions. For example, the Web Analyst version allows for the creation of derived metrics only if the Intelligence Server license includes OLAP Services.
MicroStrategy Web MMT for MicroStrategy Web

MicroStrategy Web MMT is a Web-based training module that gives you the in-depth knowledge and hands-on experience you require to work with MicroStrategy Web or Web Universal.

In previous releases, MicroStrategy Web MMT was referred to as MicroStrategy eTrainer.

Each lesson of the Web MMT course describes a function of MicroStrategy Web and breaks it into a combination of informational screens and task-based tests. Each lesson has an appropriate and graded activity. This training gives you the option to pick and choose topics that enhance your performance, ultimately leading to the success of the business intelligence or e-business project. With the included database feature, managers can control access to the training and monitor progress and achievement of users going through the Web MMT module to learn about MicroStrategy Web or Web Universal.

For more information, refer to the MicroStrategy readme.

MicroStrategy Portlets

Though different portal products typically require different integration approaches, you can integrate MicroStrategy content and functionality into your portal using one of the out-of-the-box MicroStrategy Portlets. Each out-of-the-box MicroStrategy Portlet provides a full complement of portlet features that are not found in any single portal server product, and combines the most useful features of the portlet mechanisms currently available. These portlets are designed to take advantage of the storage and repository mechanisms of its particular portal product, without requiring users to make any adjustments or changes when implementing portlets within a portal.

MicroStrategy Portlets can embed folders, reports, documents, user History Lists, and a search page into the portals through easy-to-configure screens. The portlets provide the full range of OLAP manipulations such as sort, pivot, add subtotals, export, and add new calculations, as well as design functionalities such as changing the report display between grids and graphs, and toggling thresholds.

In portal environments, users are commonly already logged in and authenticated with the portal. This authentication can also be used to provide access to MicroStrategy Web within the portal without having to
re-enter their login information. This process is known as single sign-on. Out-of-the-box MicroStrategy Portlets automatically include support for single sign-on.

For information on how install and configure out-of-the-box MicroStrategy Portlets for several major enterprise portal servers, see the Enterprise Portal Integration Help. This help documentation is installed in the Documentation folder of the MicroStrategy Common Files. Using a web browser, open the index.htm file in the default folder location:

- **32-bit Windows environments:** C:\Program Files\Common Files\MicroStrategy\Documentation\Portlets
- **64-bit Windows environments:** C:\Program Files (x86)\Common Files\MicroStrategy\Documentation\Portlets
- **UNIX or Linux:** /opt/MicroStrategy/Documentation/Portlets, or $HOME/MicroStrategy/install/Documentation/Portlets if you do not have write access to /opt/MicroStrategy.

**MicroStrategy GIS Connectors**

MicroStrategy Geospatial Information System (GIS) Connectors let you integrate with ESRI to create sophisticated GIS applications. GIS lets business users visualize data in forms such as maps, globes, reports, and charts so that they can identify and analyze relationships, patterns, and trends in their data.

For information on how install and configure the MicroStrategy GIS Connectors, see the GIS Integration Help. This help documentation is installed in the Documentation folder of the MicroStrategy Common Files. Using a web browser, open the GIS_Integration.htm file in the default folder location:

- **32-bit Windows environments:** C:\Program Files\Common Files\MicroStrategy\Documentation\GISConnectors
- **64-bit Windows environments:** C:\Program Files (x86)\Common Files\MicroStrategy\Documentation\GISConnectors
- **UNIX or Linux:** /opt/MicroStrategy/Documentation/GISConnectors, or $HOME/MicroStrategy/install/Documentation/GISConnectors if you do not have write access to /opt/MicroStrategy.
MicroStrategy Mobile

MicroStrategy Mobile is an interactive interface of the MicroStrategy BI platform that lets mobile business users harness the analytical power of MicroStrategy through the use of their BlackBerry, iPhone, and iPad devices:

- **MicroStrategy Mobile for BlackBerry**: MicroStrategy Mobile for BlackBerry lets mobile business users run reports and dashboards directly from their BlackBerry® mobile devices from Research in Motion.

  With MicroStrategy Mobile, business users receive the same reports on their BlackBerry smartphones as they do via MicroStrategy Web, MicroStrategy Office, or MicroStrategy Desktop, without the need for reformatting or retrofitting existing reports. The report manipulation features available in MicroStrategy Mobile allow users to view even the largest reports within the compact screen size of the BlackBerry smartphone.

  Designed for all levels of users, MicroStrategy Mobile reports are displayed in an easy-to-view and easy-to-navigate format, enabling users to review, analyze, and interact with important data at their convenience. Users can also access reports while they are offline, or when their BlackBerry is not within connectivity range. MicroStrategy Mobile uses the same robust security features of the MicroStrategy platform as well as the security infrastructure provided by the BlackBerry Enterprise Server.

- **MicroStrategy Mobile for iPhone**: MicroStrategy Mobile for iPhone and the MicroStrategy Mobile Server provide MicroStrategy reporting and analysis capabilities on iPhone devices. MicroStrategy utilizes iPhone’s intuitive interface to let users explore information using touch and smart gestures. MicroStrategy Mobile Business Intelligence applications can support workflows that lead users through data to decisions.

  MicroStrategy Mobile for iPhone also provides application developers a new way to develop and deploy iPhone applications that is faster, easier, and more maintainable than using traditional Integrated Development Environments. MicroStrategy Mobile for iPhone offers the following benefits:

  - **Reduces the time to develop new iPhone applications**: MicroStrategy’s Mobile application platform includes the infrastructure needed to support each new iPhone application, so that application developers only need to focus on creating the user experience and not on the back-end infrastructure.
Easy for non-developers to create professional iPhone applications: MicroStrategy’s iPhone applications do not require any coding. Using MicroStrategy's Mobile application platform, iPhone applications are assembled in a point-and-click fashion. Application designers can choose from an array of finely-designed, iPhone-optimized displays and controls.

Easy for companies to rapidly deploy iPhone application updates: MicroStrategy’s Mobile application platform uses an on-demand form of application deployment called “in-stream” deployment. As soon as new or updated applications are ready, they are instantly available to iPhone users directly from MicroStrategy’s Mobile application platform.

• MicroStrategy Mobile for iPad: MicroStrategy Mobile for iPad and the MicroStrategy Mobile Server provide MicroStrategy reporting and analysis capabilities on iPad devices. MicroStrategy utilizes iPad’s stunning touchscreen and gesture motif to allow anyone to find information naturally and intuitively.

To learn more about MicroStrategy Mobile, see the MicroStrategy Mobile Design and Administration Guide and the MicroStrategy Mobile Analysis Guide.

If you plan to use IIS 6 on 64-bit Windows operating systems, MicroStrategy Mobile must be configured as a 64-bit application. These requirements are described in Supporting MicroStrategy products with IIS 6 on 64-bit Windows operating systems, page 70.

MicroStrategy SDK

The MicroStrategy SDK is a collection of programming tools, utilities, documentation, and libraries of functions or classes that are designed to allow users to customize and extend MicroStrategy products and to integrate them within other applications. The programming tools provided by the MicroStrategy SDK— including programming instructions, points of access, and guidelines for developers— allow programmers to enhance the operation of their software by customizing and embedding the robust functionality of the MicroStrategy BI platform.

The MicroStrategy SDK provides access to the entire MicroStrategy platform and includes all of the services and utilities required for building a robust, feature-filled business intelligence-enabled application. The MicroStrategy SDK is made up of the following components:
• The MicroStrategy SDK includes the following individual SDKs, which are described in detail in the MicroStrategy Developer Library:
  ▪ Web SDK.
  ▪ Visualization SDK.
  ▪ Mobile SDK.
  ▪ Web Services SDK.
  ▪ Narrowcast Server SDK.
  ▪ Intelligence Server SDK.
  ▪ MicroStrategy Office SDK.
• Each of the individual SDKs listed above is made up of some or all of the following specific components:
  ▪ A comprehensive set of APIs that includes:
    – COM-based client-server API.
    – XML-based Web API with support for Java/COM.
    – Web Services API.
    – Narrowcast Server API.
  The set of MicroStrategy APIs provides support for a variety of development environments, including Java, C++, VB, XML, and standard Web and client-server technologies.
  ▪ A complete set of SDK documentation for all the MicroStrategy products that includes:
    – Reference guides such as Javadocs for the APIs.
    – The MicroStrategy Developer Library (MSDL), which provides all the information required to understand and use the MicroStrategy SDK.
  ▪ A variety of development tools that include:
    – Source code and sample application code for typical customization tasks.
    – Development tools and production-ready utilities that reduce code creation and maintenance and help you build customized applications.
– Specialized development tools, such as the Portal Integration Kit and the Web Services Development Kit.

– Features for packaging your application, including embedded (silent) installation, project mover for project maintenance and upgrade, and schema services to upgrade the metadata

MicroStrategy Integrity Manager

MicroStrategy Integrity Manager is an automated report comparison tool. Report SQL, report data, and graphs can be executed and compared in Integrity Manager to help customers verify change success. In addition, the report comparison output can be analyzed at the report level in MicroStrategy Integrity Manager, and as HTML and XML summary files that are generated to provide easily distributed results to other users.

To learn more about MicroStrategy Integrity Manager, see the MicroStrategy System Administration Guide.

MicroStrategy Office

MicroStrategy Office lets every Microsoft Office user run, edit, and format any MicroStrategy report directly from within Microsoft applications such as Excel, PowerPoint, and Word. MicroStrategy Office is designed using Microsoft .NET technology and accesses the MicroStrategy business intelligence platform using XML and MicroStrategy Web Services.

MicroStrategy Office gives business users open and straightforward access to the full functionality of the MicroStrategy platform from familiar Microsoft Office applications. MicroStrategy Office serves as a Microsoft add-in, with MicroStrategy functionality exposed as a single toolbar in Microsoft Office applications.


MicroStrategy Office requires that MicroStrategy Web Services is also installed. For information on Web Services, see MicroStrategy Web Services (ASP.NET) and Web Services (J2EE), page 16.
Allowing users to install MicroStrategy Office from a network location

You can allow users to install MicroStrategy Office from a network location, as described in the procedure below.

To allow users to install MicroStrategy Office from a network location

1. Insert the MicroStrategy installation disk into the disk drive and close the MicroStrategy Main Menu window that opens automatically.
2. Browse to the Installations folder on the MicroStrategy installation disk.
3. Copy the Office folder and paste it to a network location of your choice.
   
   To ensure that ASP.NET Framework and Web Services Enhancements (WSE) Runtime are installed on users’ machines when they install MicroStrategy Office, copy the Utilities folder to the network location so that it is on the same folder level as the Office folder. ASP.NET Framework and WSE Runtime are installed only if they are not already installed on the user’s machine.
4. Share the network location with any users who need to install MicroStrategy Office.
5. Notify MicroStrategy Office users to run MicroStrategyOffice.msi from within the Office folder to install MicroStrategy Office. These users will need Microsoft Windows Installer 4.5 on their machine to install MicroStrategy Office.

MicroStrategy Web Services (ASP.NET) and Web Services (J2EE)

MicroStrategy Web Services (ASP.NET) and Web Services (J2EE) are two options to support the use of MicroStrategy Office.

- MicroStrategy Web Services (ASP.NET) is an easy-to-deploy service. You can deploy the ASP.NET version using Microsoft IIS on a Windows environment.
• MicroStrategy Web Services (J2EE) provides a servlet-based version of MicroStrategy Web Services that is compatible with a UNIX, Linux, or Windows environment.

To support alternative ways to access the MicroStrategy business intelligence platform using the latest web services technologies such as ASP.NET, JNI, Java and Web protocols, such as Apache Axis, refer to the MicroStrategy SDK and the accompanying MSDL.

For information on deploying MicroStrategy Web Services ASP.NET and J2EE versions, refer to Chapter 7, Deploying MicroStrategy Web Services.

For more information on MicroStrategy Web Services (ASP.NET) and Web Services (J2EE), refer to the MicroStrategy Web Services Administration Guide.

MicroStrategy Narrowcast Server

MicroStrategy Narrowcast Server proactively distributes personalized information to employees, business partners, and customers through a variety of devices, including mobile phones, PDAs, email, Web pages, and pagers. The distribution of personalized messages and targeted offers is triggered according to predefined schedules and exception criteria, delivering information in a timely and convenient manner. Narrowcast Server also provides a self-subscription portal, easing administrative responsibilities and empowering information consumers to choose the information they receive. Narrowcast Server can draw information from relational or non-relational sources.

Subscriptions can also be supported through Intelligence Server with the introduction of Distribution Services. For information on Distribution Services, see MicroStrategy Distribution Services, page 7.

For information on Narrowcast Server subcomponents, refer to the MicroStrategy Narrowcast Server Installation and Configuration Guide.

SequeLink ODBC Socket Server

SequeLink is a complete, end-to-end solution for configuring and managing data access across virtually any number of data stores, operating systems, and deployment options. It provides a universal client for ODBC and JDBC,
and a centralized console for configuring and managing data access from a single point of control. SequeLink creates two NT services: SLAgent 54, which is the Administrator, and SLSocket54, which is the Server.

SequeLink ODBC Socket Server is installed automatically if you install MicroStrategy Narrowcast Server. You do not need to install this if you are not installing Narrowcast Server.

**MicroStrategy Analytics Modules**

MicroStrategy Analytics Modules are a set of packaged analytic components built using the MicroStrategy platform. The modules can be mapped to a different warehouse or used as starter kits to develop custom applications. Each module consists of a MicroStrategy project in a metadata, a reference guide, and a default data model.

Examples of the Analytics Modules are as follows:

- Customer Analysis Module— focuses on customer-centric information
- Sales Analysis Module— focuses on information captured during the sales process
- Financial Reporting Analysis Module— provides in-depth analysis of financial information captured by general ledgers, forecasting systems, and data on actual amounts, payables, and receivables

**MicroStrategy Tutorial Reporting**

MicroStrategy Tutorial Reporting is a sample MicroStrategy project with a metadata and warehouse, and a set of demonstration applications designed to illustrate the platform's rich functionality.

The theme of the Tutorial project is a retail store that sells electronics, books, movies, and music. The key features include:

- Five hierarchies: Customer, Geography, Products, Promotions, and Time. Each hierarchy can be viewed graphically through MicroStrategy Desktop and MicroStrategy Web (through documents).
- A large number of customers and items purchased.
- Five reporting areas: Human Resources, Inventory, Financial, Product Sales, and Supplier.
• Options to create reports from MicroStrategy Web or Desktop focusing on a particular analysis area, such as Customer, Inventory, Time, Products, Category, Employee, or Call Center.

For more information on the Tutorial project, refer to the Project Design Guide.

MicroStrategy Health Center

MicroStrategy Health Center can help you diagnose and fix problems in your MicroStrategy system. It detects known problems and provides an immediate solution. In cases where Health Center cannot fix a problem immediately, it enables you to bundle relevant log files into a diagnostic package and transmit this package to MicroStrategy Technical Support for review and troubleshooting.

Health Center is provided with a MicroStrategy installation.

For information on using Health Center to diagnose and fix problems in your MicroStrategy environment, refer to the System Administration Guide.

Installation prerequisites

Before you install MicroStrategy, you must have the following:

• MicroStrategy installation files

You can access the installation files from the MicroStrategy disk or you can ask your system administrator to share the files on a network location. For information on mounting or unmounting CD-ROM drives, see Mounting and unmounting CD-ROMs, page 526, in Appendix B, Troubleshooting.

• License key from MicroStrategy

Before you begin upgrading any MicroStrategy systems, contact your MicroStrategy account executive to obtain a new license key for the version of software you are installing. To install MicroStrategy products, you must first obtain a license key specific to the version of MicroStrategy software you intend to install.

• Installation location for your MicroStrategy products
To install MicroStrategy, you must have the following permissions and privileges:

- **Windows:**
  - You must log on to your machine using a domain account with Windows administrative privileges for the domain or target machine.
  - The user installing MicroStrategy needs write permissions in the installation directory to complete the installation; otherwise the installation fails

- **UNIX and Linux:**
  - You need root access permissions for installation if you have purchased the CPU-based MicroStrategy license.

Apart from the information provided above, review the following sections before you start the installation:

- *Recommended installation location and example deployments, page 20*
- *Hardware requirements and recommendations, page 23*
- *Software requirements and recommendations, page 34*

## Recommended installation location and example deployments

There are a countless number of possible arrangements for all the products available on the MicroStrategy platform, and what you decide to do depends largely on your installation environment and requirements. In general, though, the following recommendations are usually true for a typical business intelligence system:

- Intelligence Server should be installed on its own dedicated server machine.
- MicroStrategy Web should be installed on its own dedicated Web server machine.
- The rest of the products can be installed in varying combinations depending on who intends to use them and on what machines.

These are just suggestions to help you get started. Read the rest of this chapter for more detailed guidelines.
The following sections provide basic examples of differently sized production deployments with MicroStrategy products. The examples are generalized and do not include all of the MicroStrategy products. You can use these examples to help plan how to deploy MicroStrategy products.

**Small production deployment**

500 Web Users

50 MicroStrategy Office Users

100 E-mail, File/Print Server Users

MicroStrategy Web and Web Services

MicroStrategy Narrowcast server

MicroStrategy Intelligence Server
MicroStrategy OLAP Services
MicroStrategy Report Services
MicroStrategy Distribution Services

5 Users

MicroStrategy Desktop

2 Users

MicroStrategy Architect
Medium production deployment

5,000 Web Users

250 MicroStrategy Office Users

10,000 E-mail, File/Print Server Users

50 Mobile Users

MicroStrategy Web

MicroStrategy Web Services

MicroStrategy Narrowcast Server

MicroStrategy Mobile Server

MicroStrategy Desktop

MicroStrategy Architect

MicroStrategy Intelligence Server
MicroStrategy OLAP Services
MicroStrategy Report Services
MicroStrategy Distribution Services

MicroStrategy Command Manager
MicroStrategy Enterprise Manager
MicroStrategy Object Manager
Large production deployment

Hardware requirements and recommendations

MicroStrategy acknowledges that variables, such as CPU speed, CPU type, operating system version, service upgrades, file space, and physical and swap memory, are factors that play an important role in making your deployment of MicroStrategy a successful one. This section lists the hardware requirements and recommendations for Windows, UNIX, and Linux platforms.

The following information is intended to give you general guidance on hardware requirements to support the MicroStrategy product suite. Determining the necessary hardware requirements to support MicroStrategy is dependent on many factors including but not limited to the complexity of your MicroStrategy environment, the deployment strategy of MicroStrategy features, user community requirements, expected peak usage requirements, and response time expectations. Factors such as these must be considered to determine the hardware requirements for your MicroStrategy production environment.
For more details and exact information regarding supported and certified operating system versions for a particular MicroStrategy release, see the readme files or contact Technical Support.

For UNIX and Linux systems, a number of system settings can affect the performance of MicroStrategy Intelligence Server Universal. These settings do not need to be set prior to a MicroStrategy installation. For more information on these settings and their recommended values, see Recommended system settings for UNIX and Linux, page 82.

**System hardware requirements and recommendations for Windows**

The following table shows the recommended and minimum hardware requirements for MicroStrategy products. See the products’ readme files for the updated requirements.

<table>
<thead>
<tr>
<th>MicroStrategy Product</th>
<th>Processor</th>
<th>Memory</th>
<th>Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>MicroStrategy Command Manager</td>
<td>x86 or x64 compatible</td>
<td>2 GB (minimum 512 MB)</td>
<td>1 GB</td>
</tr>
<tr>
<td>MicroStrategy Enterprise Manager</td>
<td>x86 or x64 compatible</td>
<td>1 GB (minimum 512 MB)</td>
<td>1 GB</td>
</tr>
<tr>
<td>MicroStrategy Object Manager</td>
<td>x86 or x64 compatible</td>
<td>1 GB (minimum 512 MB)</td>
<td>1 GB</td>
</tr>
<tr>
<td>MicroStrategy Desktop products</td>
<td>x86 compatible</td>
<td>2 GB or higher</td>
<td>1 GB</td>
</tr>
<tr>
<td>MicroStrategy Intelligence Server</td>
<td>x86 compatible</td>
<td>4 GB or higher</td>
<td>4 GB</td>
</tr>
</tbody>
</table>

The following information is intended to give you general guidance on hardware requirements to support the MicroStrategy product suite. Determining the necessary hardware requirements to support MicroStrategy is dependent on many factors including but not limited to the complexity of your MicroStrategy environment, the deployment strategy of MicroStrategy features, user community requirements, expected peak usage requirements, and response time expectations. Factors such as these must be considered to determine the hardware requirements for your MicroStrategy production environment.
### Planning Your Installation

<table>
<thead>
<tr>
<th>MicroStrategy Product</th>
<th>Processor</th>
<th>Memory</th>
<th>Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>MicroStrategy Intelligence Server Universal</td>
<td>x64 compatible</td>
<td>4 GB or higher</td>
<td>Three times the amount of RAM available to Intelligence Server. For example, an Intelligence Server that is provided 4 GB of RAM requires 12 GB of hard drive space.</td>
</tr>
<tr>
<td>MicroStrategy Integrity Manager</td>
<td>x86 or x64 compatible</td>
<td>2 GB or higher</td>
<td>2 GB (minimum 1 GB)</td>
</tr>
<tr>
<td>MicroStrategy Office</td>
<td>x86 or x64 compatible</td>
<td>2 GB (minimum 1 GB)</td>
<td>1 GB</td>
</tr>
<tr>
<td>MicroStrategy Mobile Server</td>
<td>x86 or x64 compatible</td>
<td>1 GB (minimum 512 MB)</td>
<td>2 GB (minimum 1 GB)</td>
</tr>
<tr>
<td>MicroStrategy Narrowcast Server</td>
<td>x86 or x64 compatible</td>
<td>1 GB (minimum 512 MB)</td>
<td>2 GB (minimum 1 GB)</td>
</tr>
<tr>
<td>MicroStrategy SDK</td>
<td>N/A</td>
<td>N/A</td>
<td>300 MB</td>
</tr>
<tr>
<td>MicroStrategy Tutorial</td>
<td>x86 or x64 compatible</td>
<td>1 GB (minimum 512 MB)</td>
<td>1 GB</td>
</tr>
<tr>
<td>MicroStrategy Analytics Modules</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>MicroStrategy Web: Web Server</td>
<td>x86 compatible</td>
<td>4 GB or higher</td>
<td>1 GB</td>
</tr>
<tr>
<td>MicroStrategy Web: Web Client</td>
<td>x86 or x64 compatible</td>
<td>2 GB or higher</td>
<td>Additional space not required</td>
</tr>
</tbody>
</table>
In addition to the MicroStrategy products listed above, the database server on which your MicroStrategy metadata is hosted should run on a computer with the following minimum configuration:

- Processor: x86 or x64 compatible
- Memory: 256 MB
- Storage: 200 MB

Be aware of the following:

- Depending on the MicroStrategy products you intend to install, you must have at least 400 MB to 830 MB of free space for a set of common files that MicroStrategy installs. If you only install MicroStrategy Web, you need 400 MB of free space. However, the common files for the entire MicroStrategy product suite requires 830 MB of free space. You can store these files in a directory of your choice during an installation. With a typical installation setup type, these files are installed on the C: drive. This storage requirement is in addition to the storage requirement discussed in the above list.

- Intelligence Server is licensed based on CPU number and clock speed. Thus, Intelligence Server can only be installed on machines with a maximum clock speed that equals the licensed clock speed. If you try to install the product on a machine faster than what is licensed, installation fails. Additionally, you must have an Intelligence Server Universal license to install on 64-bit Windows operating systems. See the MicroStrategy System Administration Guide for more information about licensing.
Intelligence Server Universal hardware requirements on UNIX/Linux

For the exact information, such as version numbers and space requirements, see the readme file.

The following information is intended to give you general guidance on hardware requirements to support the MicroStrategy product suite. Determining the necessary hardware requirements to support MicroStrategy is dependent on many factors including but not limited to the complexity of your MicroStrategy environment, the deployment strategy of MicroStrategy features, user community requirements, expected peak usage requirements, and response time expectations. Factors such as these must be considered to determine the hardware requirements for your MicroStrategy production environment.

Before you review the sections below, be aware that a successful configuration of Intelligence Server Universal depends on a valid combination of an operating system and a CPU architecture. For example, Intelligence Server Universal is certified to run on a machine with an Oracle Solaris operating system compatible with a SPARC CPU architecture.

Valid operating system and CPU architecture combinations for Intelligence Server Universal are listed in the sections below.
Oracle Solaris

Hardware requirements

<table>
<thead>
<tr>
<th>CPU</th>
<th>Memory</th>
<th>File Space</th>
<th>Disk Drive</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPARC platforms</td>
<td>4 GB</td>
<td>Three times the amount of RAM available to Intelligence Server. For example, an Intelligence Server that is provided 4 GB of RAM requires 12 GB of hard drive space.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

CPU: The use of 4 GB of RAM is a minimum level of support for the MicroStrategy Product Suite, which does not take into account the performance of a production system. Performance testing has shown that 64 GB or more of RAM should be available to allow MicroStrategy Intelligence Server Universal to fully support and take advantage of the complete feature set of the MicroStrategy Product Suite, while obtaining system-wide high performance. This level of system resources allows MicroStrategy Intelligence Server Universal to fully utilize performance-improving technologies such as MicroStrategy OLAP Services, and to support optimal performance for MicroStrategy Report Services documents and dashboards and the other features of the MicroStrategy Product Suite.

Space requirements on Oracle Solaris

Common files referred to in this table are installed no matter which MicroStrategy products are selected.

<table>
<thead>
<tr>
<th>Directory</th>
<th>Common Files</th>
<th>Intelligence Server</th>
<th>Web Universal</th>
<th>SDK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Directory</td>
<td>427 KB</td>
<td>107 KB</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Install Directory</td>
<td>260 MB</td>
<td>340 MB</td>
<td>50 MB</td>
<td>0</td>
</tr>
<tr>
<td>Log Directory</td>
<td>2 KB</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Web Universal Deploy Directory</td>
<td>N/A</td>
<td>N/A</td>
<td>26.4 MB</td>
<td>N/A</td>
</tr>
<tr>
<td>SDK Install Directory</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>281 MB</td>
</tr>
</tbody>
</table>

In addition to the space requirements listed above, you also need 370 MB of free space in the temporary directory. The default location of the temporary directory is /var/tmp.

If the space in the default temporary directory is inadequate, you can use the is:tempdir command to change the location of the temporary directory.
This directory must already exist and it must be specified using its absolute path, for example:

```
./setupsol.bin -is:tempdir /home/user/tmp
```

If you change the location of the temporary directory, 150 KB of free space is still required in the `/var/tmp` directory to launch the installation routine.

**IBM AIX**

**Hardware requirements**

<table>
<thead>
<tr>
<th>CPU</th>
<th>Memory</th>
<th>File Space</th>
<th>Disk Drive</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM POWER</td>
<td>4 GB</td>
<td>Three times the amount of RAM available to Intelligence Server. For example, an Intelligence Server that is provided 4 GB of RAM requires 12 GB of hard drive space.</td>
<td>Yes</td>
</tr>
<tr>
<td>Architecture</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The use of 4 GB of RAM is a minimum level of support for the MicroStrategy Product Suite, which does not take into account the performance of a production system. Performance testing has shown that 64 GB or more of RAM should be available to allow MicroStrategy Intelligence Server Universal to fully support and take advantage of the complete feature set of the MicroStrategy Product Suite, while obtaining system-wide high performance. This level of system resources allows MicroStrategy Intelligence Server Universal to fully utilize performance-improving technologies such as MicroStrategy OLAP Services, and to support optimal performance for MicroStrategy Report Services documents and dashboards and the other features of the MicroStrategy Product Suite.</td>
<td></td>
</tr>
</tbody>
</table>

The AIX kernel should be running only in 64-bit mode. You can determine the mode by logging in as the root user and entering `bootinfo -K` at the command prompt. The output displays the mode.

**Space requirements on IBM AIX**

Common Files referred to in this table are installed no matter which MicroStrategy products are selected.

<table>
<thead>
<tr>
<th>Directory</th>
<th>Common Files</th>
<th>Intelligence Server</th>
<th>Web Universal</th>
<th>SDK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Directory</td>
<td>551 KB</td>
<td>103 KB</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Install Directory</td>
<td>342 MB</td>
<td>1009 MB</td>
<td>60 MB</td>
<td>0</td>
</tr>
</tbody>
</table>
In addition to the space requirements listed above, you also need 320 MB of free space in the temporary directory. The default location of the temporary directory is /tmp.

If the space in the default temporary directory is inadequate, you can use the `is:tempdir` command to change the location of the temporary directory. This directory must already exist and it must be specified using its absolute path, for example:

```
./setupAIX.bin -is:tempdir /home/user/tmp
```

If you change the location of the temporary directory, 150 KB of free space is still required in the /tmp directory to launch the installation routine.

### HP-UX

#### Hardware requirements

<table>
<thead>
<tr>
<th>CPU</th>
<th>Memory</th>
<th>File Space</th>
<th>Disk Drive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel Itanium</td>
<td>4 GB</td>
<td>Three times the amount of RAM available to Intelligence Server. For example, an Intelligence Server that is provided 4 GB of RAM requires 12 GB of hard drive space.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

The use of 4 GB of RAM is a minimum level of support for the MicroStrategy Product Suite, which does not take into account the performance of a production system. Performance testing has shown that 64 GB or more of RAM should be available to allow MicroStrategy Intelligence Server Universal to fully support and take advantage of the complete feature set of the MicroStrategy Product Suite, while obtaining system-wide high performance. This level of system resources allows MicroStrategy Intelligence Server Universal to fully utilize performance-improving technologies such as MicroStrategy OLAP Services, and to support optimal performance for MicroStrategy Report Services documents and dashboards and the other features of the MicroStrategy Product Suite.
Space requirements on HP-UX

Common Files referred to in this table are installed no matter which MicroStrategy products are selected.

<table>
<thead>
<tr>
<th>Directory</th>
<th>Common Files</th>
<th>Intelligence Server</th>
<th>Web Universal</th>
<th>SDK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Directory</td>
<td>400 KB</td>
<td>85 KB</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Install Directory</td>
<td>490 MB</td>
<td>302 MB</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Log Directory</td>
<td>2 KB</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Web Universal Deploy Directory</td>
<td>N/A</td>
<td>N/A</td>
<td>26.8 MB</td>
<td>N/A</td>
</tr>
<tr>
<td>SDK Install Directory</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>234 MB</td>
</tr>
</tbody>
</table>

In addition to the space requirements listed above, you also need 780 MB of free space in the temporary directory. The default location of the temporary directory is /tmp. If the space in the default temporary directory is inadequate, you can use the is:tempdir command to change the location of the temporary directory. This directory must already exist and it must be specified using its absolute path, for example:

```
./setupHPIA64.bin -is:tempdir /home/user/tmp
```

Red Hat and SUSE Linux

Hardware requirements

<table>
<thead>
<tr>
<th>CPU</th>
<th>Memory</th>
<th>File Space</th>
<th>Disk Drive</th>
</tr>
</thead>
<tbody>
<tr>
<td>x86-64 compatible</td>
<td>4 GB</td>
<td>Three times the amount of RAM available to Intelligence Server. For example, an Intelligence Server that is provided 4 GB of RAM requires 12 GB of hard drive space.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The use of 4 GB of RAM is a minimum level of support for the MicroStrategy Product Suite, which does not take into account the performance of a production system. Performance testing has shown that 64 GB or more of RAM should be available to allow MicroStrategy Intelligence Server Universal to fully support and take advantage of the complete feature set of the MicroStrategy Product Suite, while obtaining system-wide high performance. This level of system resources allows MicroStrategy Intelligence Server Universal to fully utilize performance-improving technologies such as MicroStrategy OLAP Services, and to support optimal performance for MicroStrategy Report Services documents and dashboards and the other features of the MicroStrategy Product Suite.</td>
<td></td>
</tr>
</tbody>
</table>
Space requirements for Red Hat and SUSE Linux

Common Files referred to in this table are installed no matter which MicroStrategy products are selected.

<table>
<thead>
<tr>
<th>Directory</th>
<th>Common Files</th>
<th>Intelligence Server</th>
<th>Web Universal</th>
<th>SDK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Directory</td>
<td>425 KB</td>
<td>112 KB</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Install Directory</td>
<td>314.8 MB</td>
<td>465.2 MB</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Log Directory</td>
<td>2 KB</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Web Universal Deploy Directory</td>
<td>N/A</td>
<td>N/A</td>
<td>27.2 KB</td>
<td>N/A</td>
</tr>
<tr>
<td>SDK Install Directory</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>243.9 MB</td>
</tr>
</tbody>
</table>

In addition to the space requirements listed above, you also need 202 MB of free space in the temporary directory. The default location of the temporary directory is /tmp.

If the space in the default temporary directory is inadequate, you can use the is:tempdir command to change the location of the temporary directory. This directory must already exist and it must be specified using its absolute path, for example:

```
./setupLinux.bin -is:tempdir /home/user/tmp
```

⚠️ If you change the location of the temporary directory, 150 KB of free space is still required in the /tmp directory to launch the installation routine.

MicroStrategy Web Universal hardware requirements and recommendations on UNIX/Linux

To verify updated requirement information, see the MicroStrategy readme.

⚠️ The following information is intended to give you general guidance on hardware requirements to support the MicroStrategy product suite. Determining the necessary hardware requirements to support MicroStrategy is dependent on many factors including but not limited to the complexity of your MicroStrategy environment, the deployment strategy of MicroStrategy features, user community requirements, expected peak usage requirements, and response time expectations. Factors such as these must be considered to determine the hardware requirements for your MicroStrategy production environment.
Web server

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Minimum</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>4 GB or higher recommended per CPU core</td>
</tr>
<tr>
<td>Hard disk space</td>
<td>1 GB or higher</td>
</tr>
</tbody>
</table>

Web client

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Minimum</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>2 GB or higher</td>
</tr>
</tbody>
</table>

MicroStrategy Mobile hardware requirements for BlackBerry devices

The tables below list the MicroStrategy Mobile client application hardware requirements for BlackBerry, iPhone, iPod Touch, and iPad devices. To verify updated requirement information, see the MicroStrategy readme.

Flash memory

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Minimum</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flash memory</td>
<td>32 MB</td>
<td>64 MB</td>
</tr>
</tbody>
</table>

BlackBerry devices

<table>
<thead>
<tr>
<th>BlackBerry devices</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlackBerry Torch 9800 Series</td>
<td>Supported</td>
</tr>
<tr>
<td>BlackBerry Bold 9700 Series</td>
<td>Certified</td>
</tr>
<tr>
<td>BlackBerry Style 9670</td>
<td>Supported</td>
</tr>
<tr>
<td>BlackBerry Tour 9630</td>
<td>Supported</td>
</tr>
<tr>
<td>BlackBerry Pearl 3G 9100 Series</td>
<td>Supported</td>
</tr>
<tr>
<td>BlackBerry Bold 9000 Series</td>
<td>Certified</td>
</tr>
<tr>
<td>BlackBerry Curve 8900 Series</td>
<td>Supported</td>
</tr>
<tr>
<td>BlackBerry 8800 Series</td>
<td>Certified</td>
</tr>
</tbody>
</table>
Installation prerequisites © 2011 MicroStrategy, Inc.

### BlackBerry devices

<table>
<thead>
<tr>
<th>Devices</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlackBerry 8700 Series</td>
<td>Certified</td>
</tr>
<tr>
<td>BlackBerry Curve 8500 Series</td>
<td>Supported</td>
</tr>
<tr>
<td>Blackberry Curve 8300 Series</td>
<td>Certified</td>
</tr>
<tr>
<td>BlackBerry Pearl Flip 8200 Series</td>
<td>Supported</td>
</tr>
<tr>
<td>BlackBerry Pearl 8100 Series</td>
<td>Certified</td>
</tr>
<tr>
<td>BlackBerry 7200 Series</td>
<td>Certified</td>
</tr>
<tr>
<td>BlackBerry 7100 Series</td>
<td>Certified</td>
</tr>
</tbody>
</table>

### iPhone and iPod Touch devices

<table>
<thead>
<tr>
<th>Devices</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPhone 4</td>
<td>Certified</td>
</tr>
<tr>
<td>iPhone 3G</td>
<td>Certified</td>
</tr>
<tr>
<td>iPhone 3GS</td>
<td>Certified</td>
</tr>
<tr>
<td>iPod Touch</td>
<td>Certified</td>
</tr>
</tbody>
</table>

### iPad devices

<table>
<thead>
<tr>
<th>Devices</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPad Wi-Fi</td>
<td>Certified</td>
</tr>
<tr>
<td>iPad Wi-Fi + 3G</td>
<td>Certified</td>
</tr>
<tr>
<td>iPad 2 Wi-Fi</td>
<td>Certified</td>
</tr>
<tr>
<td>iPad 2 Wi-Fi + 3G</td>
<td>Certified</td>
</tr>
</tbody>
</table>

## Software requirements and recommendations

This section lists the software requirements and recommendations for MicroStrategy products on the Windows, UNIX, and Linux platforms.

The following information is intended to give you general guidance on software requirements. For details regarding supported versions of the operating system for a particular MicroStrategy release, see the readme files or contact Technical Support.
System software requirements and recommendations on Windows

For a successful installation of MicroStrategy products, the following product-specific software requirements must be satisfied. For more detailed information on product-specific requirements, see the products’ readme files.

Be aware of the following before reviewing the sections below:

- The operating systems listed below are deemed supported or certified to reflect the level of internal testing that each configuration was exposed to for the current release. MicroStrategy recommends using certified configurations over the supported configurations.

- MicroStrategy certifies and supports operating systems that are compatible with a set of CPU chipsets, referred to as CPU architectures, that are binary-compatible. MicroStrategy tests on at least one of the CPU chipsets within a set of binary-compatible CPU architectures for purposes of certifying and supporting operating systems with MicroStrategy products. A valid CPU architecture is provided in parentheses () to clarify the operating system software certified or supported for MicroStrategy Administrator.
**MicroStrategy product: Command Manager**

For Command Manager operating system requirements on UNIX and Linux platforms, see *MicroStrategy Command Manager software requirements on UNIX/Linux, page 66.*

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Enterprise Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Professional Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Web browser</strong></td>
</tr>
<tr>
<td></td>
<td>• Microsoft Internet Explorer 7.x, 8.x, and 9.x</td>
</tr>
<tr>
<td></td>
<td>• Firefox 3.x and 4.x</td>
</tr>
<tr>
<td></td>
<td>• Google Chrome 9.x and 10.x</td>
</tr>
<tr>
<td>Export application</td>
<td>• Microsoft Excel 2010</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Excel 2007 SP2</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Excel 2003 SP3</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Excel 2002 (XP) SP3</td>
</tr>
</tbody>
</table>
## MicroStrategy product: Desktop

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Operating system</strong></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP3 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Enterprise Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Professional Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP1 (on x64)</td>
</tr>
<tr>
<td><strong>Web browser</strong></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Microsoft Internet Explorer 7.x, 8.x, and 9.x</td>
</tr>
<tr>
<td><strong>Export application</strong></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Adobe Acrobat Reader 8.x, 9.x, and 10.x</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2002 (XP) SP3</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2003 SP3</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2007 SP2</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2010</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office for Mac 2011</td>
</tr>
<tr>
<td><strong>Export devices</strong></td>
<td>Export devices can open a PDF file created by using MicroStrategy to export an object as a PDF. You can load a PDF file exported from MicroStrategy onto an export device in a couple of different ways. You can connect the export device directly to the machine that stores the PDF to load it onto the export device. You can also email the PDF file to the email account for the export device.</td>
</tr>
<tr>
<td></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Nook firmware version 1.4</td>
</tr>
<tr>
<td></td>
<td>• Kindle firmware version 2.5.2</td>
</tr>
<tr>
<td><strong>Flash player</strong></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Adobe Flash Player 10.2</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Adobe Flash Player 10.1</td>
</tr>
</tbody>
</table>
### MicroStrategy product: Enterprise Manager

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported for demonstration purposes only:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP3 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP1 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Professional Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td>Web browser</td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Microsoft Internet Explorer 7.x</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Internet Explorer 8.x</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Internet Explorer 9.x</td>
</tr>
</tbody>
</table>
### Installation and Configuration Guide

#### Planning Your Installation

### Installation prerequisites

#### MicroStrategy product: Intelligence Server

**Intelligence Server on Windows (32-bit)**

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
</table>
| Databases (Enterprise Manager warehouse) | **Certified:**  
- IBM DB2 UDB for Linux, UNIX, & Windows 9.5 with Fix Pack 7  
- IBM DB2 UDB for Linux, UNIX, & Windows 9.7 with Fix Pack 3a  
- Microsoft SQL Server 2000 SP3/SP3a/SP4  
- Microsoft SQL Server 2005 SP1/SP2/SP3/SP4  
- Microsoft SQL Server 2008 SP1/2008 SP2/2008 R2  
- Oracle 10g and 10gR2  
  
Partitioning is supported for the Oracle Standard Edition of these database platforms.  
- Oracle 11g and 11gR2  
  
Partitioning is supported for the Oracle Standard Edition of these database platforms.  
- Sybase ASE 15  
- Sybase ASE 15.5  
- Teradata V12  
- Teradata V13  
- Teradata V13.10  
  
**Supported:**  
- IBM DB2 UDB for Linux, UNIX, & Windows 9.1 with Fix Pack 10  
- Teradata V2R6.2.x  
- Oracle 9i and 9iR2  
  
Partitioning is supported for the Oracle Standard Edition of these database platforms. |

© 2011 MicroStrategy, Inc.
**Installation and Configuration Guide**

**Installation prerequisites © 2011 MicroStrategy, Inc.**

Intelligence Server Universal on Windows (64-bit)

You must have an Intelligence Server Universal license to install on 64-bit Windows operating systems.

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory Resources</td>
<td>4GT mode is certified and recommended for the Windows version of Intelligence Server.</td>
</tr>
</tbody>
</table>

MicroStrategy Intelligence Server 9 contains new features and numerous enhancements that may require additional hardware resources when migrating from MicroStrategy 8.x production systems running on Windows 32-bit platforms.

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported for demonstration purposes only:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Professional Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Enterprise Edition SP1 (on x64)</td>
</tr>
</tbody>
</table>
## MicroStrategy product: Object Manager

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP3 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Enterprise Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Professional Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP1 (on x64)</td>
</tr>
</tbody>
</table>

| Export application     | • Microsoft Excel 2010 |
|                        | • Microsoft Excel 2007 SP2 |
|                        | • Microsoft Excel 2003 SP3 |
|                        | • Microsoft Excel 2002 (XP) SP3 |
**MicroStrategy product: MicroStrategy Office**

**MicroStrategy Office client system certified and supported configurations**

The following are supported and certified operating systems, Microsoft Office applications, and other requirements to support MicroStrategy Office client systems:

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP3 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Ultimate Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Enterprise Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Professional Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Ultimate Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP1 (on x64)</td>
</tr>
<tr>
<td>Microsoft Office (Excel, Powerpoint, and Word)</td>
<td>To successfully integrate MicroStrategy Office with your Microsoft Office products, you must use the 32-bit version of Microsoft Office products. For information on installing and supporting the 32-bit version of Microsoft Office products, refer to your third-party Microsoft documentation.</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2010 (32-bit version)</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2007 SP2 (32-bit version)</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2003 SP3 (32-bit version)</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office XP SP3 (32-bit version)</td>
</tr>
</tbody>
</table>
Installation and Configuration Guide

Planning Your Installation

Software | Software Requirements
---|---
Adobe Flash Player | To include Flash-enabled documents in Excel, PowerPoint, and Word, you must have Adobe Flash Player installed on your machine. If Flash Player is not installed, a dialog box is displayed when you execute a document. The dialog box provides a link to the Adobe download website, from which you can install Flash Player.

Certified:
• Adobe Flash Player 10.2

Supported:
• Adobe Flash Player 10.1

Microsoft .NET Framework | Microsoft .NET Framework 3.5 (encompassing .NET Framework 2.0 SP1, .NET Framework 3.0, and Microsoft VC9 runtime). If this .NET Framework is not found, it is installed as part of a platform or standalone installation.

The exceptions to this are when MicroStrategy Office is installed from MicroStrategy Web, or when upgrading MicroStrategy Office. In these scenarios, .NET Framework 3.5 must be manually installed prior to installing from MicroStrategy Web or upgrading.

Microsoft Web Services Enhancements Runtime | Microsoft Web Services Enhancements (WSE) Runtime 3.0 which is automatically installed with MicroStrategy Office if not found on the client machine.


Microsoft IIS | Microsoft Internet Information Services 6.0, 7.0, or 7.5

If you plan to use IIS 7 as the web server for MicroStrategy Web or Web Universal, you must ensure that some IIS settings are enabled, as described in Supporting IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server, page 68.

**MicroStrategy Web Services server system certified and supported configurations**

The following are supported and certified requirements to support and deploy MicroStrategy Web Services ASP.NET. For a list of software requirements and recommendations for MicroStrategy Web Services J2EE, see MicroStrategy Web Services J2EE software requirements and recommendations, page 64.
## Software Requirements

<table>
<thead>
<tr>
<th>Software</th>
<th>Operating System</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Certified:</strong></td>
<td>Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2003 Web Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2003 Standard Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2003 Enterprise Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2003 Standard Edition R2 SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2003 Enterprise Edition R2 SP2 (on x86 or x64)</td>
</tr>
<tr>
<td><strong>Supported:</strong></td>
<td>Windows XP Professional Edition SP3 (on x86)</td>
</tr>
<tr>
<td></td>
<td>Windows Vista Ultimate Edition SP1 or SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Vista Business Edition SP1 or SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>Windows 7 Professional Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>Windows 7 Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2008 Standard Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2008 Enterprise Edition SP1 (on x64)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Web server</th>
<th>Microsoft Internet Information Services 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>* If you plan to use IIS 6 on 64-bit Windows operating systems, some MicroStrategy products must be configured as 64-bit applications, and some MicroStrategy products must be configured as 32-bit applications. These requirements are described in <a href="#">Supporting MicroStrategy products with IIS 6 on 64-bit Windows operating systems, page 70</a>.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>.NET Framework</th>
<th>.NET Framework 2.0 and 3.0 (32-bit versions or 64-bit versions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Web Services</td>
<td>Microsoft Web Services Enhancements (WSE) Runtime 3.0 for</td>
</tr>
<tr>
<td>Enhancements</td>
<td>Microsoft .NET.</td>
</tr>
</tbody>
</table>
**MicroStrategy product: MicroStrategy Web**

For a list of software requirements and recommendations for MicroStrategy Web Universal, see *MicroStrategy Web Universal software requirements and recommendations, page 58.*

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td><strong>Web server:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x86) - Supported not certified</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x86) - Supported not certified</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x86) - Supported not certified</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x86) - Supported not certified</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP3 (on x86) - For demonstration purposes only</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP1 or SP2 (on x86) - For demonstration purposes only</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Professional Edition SP1 (on x86) - For demonstration purposes only</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Enterprise Edition SP1 (on x86) - For demonstration purposes only</td>
</tr>
<tr>
<td></td>
<td><strong>Web client:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows 2003 SP2 all editions (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 2003 R2 SP2 all editions (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP SP3 all editions (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP SP2 all editions (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista SP1 or SP2 all editions (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 SP1 all editions (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 SP1 all editions (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 R2 SP1 all editions (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 SP2 all editions (on x64)</td>
</tr>
<tr>
<td>Software</td>
<td>Software Requirements</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Web server          | • Microsoft Internet Information Services 6  
                        If you plan to use IIS 6 on 64-bit Windows operating systems, some MicroStrategy products must be configured as 64-bit applications, and some MicroStrategy products must be configured as 32-bit applications. These requirements are described in *Supporting MicroStrategy products with IIS 6 on 64-bit Windows operating systems, page 70.*  
                        • Microsoft Internet Information Services 7  
                        If you plan to use IIS 7 as the web server for MicroStrategy Web, you must ensure that some IIS options are enabled, as described in *Supporting IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server, page 68.*  
                        • Microsoft Internet Information Services 7.5  
                        If you plan to use IIS 7.5 as the web server for MicroStrategy Web, you must ensure that some IIS options are enabled, as described in *Supporting IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server, page 68.*  
                        • Microsoft Internet Information Services 5.1 - Supported only for demonstration purposes on 32-bit operating systems |
| Portal server       | MicroStrategy Portlets allow MicroStrategy Web to be easily integrated and configured on portal environments.  
                        **Certified:**  
                        • IBM WebSphere Portal 6.1  
                        • IBM WebSphere Portal 7  
                        • Microsoft Office SharePoint Portal 2007  
                        • Microsoft SharePoint Portal Server 2010  
                        • Oracle WebLogic Portal 10.3.2  
                        • Oracle WebLogic Portal 9.2  
                        • SAP NetWeaver Composition Environment 7.1  
                        • Liferay Portal 5.2  
                        • Liferay Portal 6.0  
                        • Drupal 6.20  
                        • DotNetNuke web content management system 5.6 |
| GIS components      | MicroStrategy GIS Connectors let you integrate with ESRI to create sophisticated GIS applications. The following GIS components are required to support this GIS integration.  
                        **Certified:**  
                        • ArcGIS 9.3.1  
                        • ArcGIS 10 |
| Web browsers (Windows client) | **Certified:**  
                        • Microsoft Internet Explorer 7.x, 8.x, and 9.x  
                        • Firefox 3.x and 4.x  
                        • Google Chrome 9.x and 10.x |
### Export application

**Certified:**
- Adobe Acrobat Reader 8.x, 9.x, and 10.x
- Microsoft Office 2002 (XP) SP3
- Microsoft Office 2003 SP3
- Microsoft Office 2007 SP2
- Microsoft Office 2010

### Export devices

Export devices can open a PDF file created by using MicroStrategy to export an object as a PDF. You can load a PDF file exported from MicroStrategy onto an export device in a couple of different ways. You can connect the export device directly to the machine that stores the PDF to load it onto the export device. You can also email the PDF file to the email account for the export device.

**Certified:**
- Nook firmware version 1.4
- Kindle firmware version 2.5.2

### .NET Framework

.NET Framework 2.0, 3.0, or 3.5 (32-bit versions)

MicroStrategy automatically installs the version of .NET required for MicroStrategy Web, if that version of .NET cannot be located on your installation machine. MicroStrategy also automatically configures the MicroStrategy Web virtual directory to run with the version of .NET that it requires.

### Adobe Flash Player

Adobe Flash Player 10.1

---

### MicroStrategy product: SDK

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
</table>
| MicroStrategy             | • Before you attempt to run the MicroStrategy SDK, you need to have a MicroStrategy project running in an Intelligence Server-based (three-tier or four-tier) environment.  
  • MicroStrategy Web or Web Universal should also be installed in addition to MicroStrategy SDK to perform custom application work on MicroStrategy Web or Web APIs. |
| .NET                      | • .NET 2.0 (32-bit or 64-bit version)  
  • .NET 3.0 (32-bit or 64-bit version)  
  • .NET 3.5 (32-bit or 64-bit version) |
| IDEs for Flex Application Development | **Certified:**  
  • Adobe Flash Builder 4.0.1 with Adobe Flex SDK 4.1  
  The Adobe Flash Builder application is not included with a MicroStrategy installation. This application is required to develop custom visualizations you can include in MicroStrategy with the use of MicroStrategy SDK. Refer to [http://www.adobe.com](http://www.adobe.com) to install the Adobe Flash Builder application. |
<table>
<thead>
<tr>
<th>Software and SDKs for MicroStrategy Mobile customization</th>
<th>Software Requirements</th>
</tr>
</thead>
</table>
| IDEs for Web customization | **Certified:**  
  - Eclipse 3.6  
  - Eclipse 3.5  
  
The Web Customization Editor is an Eclipse READY™ product that easily integrates into the Eclipse IDE. This allows you to use the Web Customization Editor as well as the complete functionality of the IDE to perform basic as well as advanced customization tasks. Refer to [http://www.eclipse.org](http://www.eclipse.org) to install the Eclipse IDE. |
| Web Services SDK components | The Web Services SDK provides a jump start to developers who want to develop and deploy Web Services-based applications. It is built using the Web SDK and makes use of the multi-layered MicroStrategy Web architecture that clearly separates functionality from presentation. |
| Creating task-based Web Services | **Certified:**  
  - Windows Server 2008 Enterprise Edition R2 (on x64)  
  - Windows Server 2003 R2 SP2 Standard Edition (on x86 or x64)  
  - Windows XP Professional Edition SP3 (on x86)  
  - Apache Ant 1.8.1  
  - Apache Axis2 1.5.2  
  - .NET SDK 2.0  
  
To support the creation of Web Services SDK customizations using the Web Customizations Editor, your system must meet the following requirements: |
| Deploying task-based Web Services | **Certified:**  
  - Windows Server 2008 Enterprise Edition R2 (on x64)  
  - Windows Server 2003 R2 SP2 Standard Edition (on x86 or x64)  
  - Windows XP Professional Edition SP3 (on x86)  
  
To support the deployment of your Web Services SDK customizations as part of your MicroStrategy Web applications, your web server system must meet the following requirements: |
| IDE and SDKs for MicroStrategy Mobile customization | The MicroStrategy Mobile SDK allows you to create customizations for MicroStrategy Mobile for iPhone and iPad. To support the creation of MicroStrategy Mobile SDK customizations, your system must meet the following requirements: |
| **Certified IDE:** |  
  - Xcode IDE 3.2.5  
  - Xcode IDE 4.0.2  
  
**Certified SDK for iPhone and iPad:**  
  - iOS SDK 4.2  
  - iOS SDK 4.3  
  
To support the creation of MicroStrategy Mobile SDK customizations, your system must meet the following requirements: |
**MicroStrategy product: Integrity Manager**

For Integrity Manager requirements on UNIX and Linux platforms, see *Intelligence Server Universal software requirements on UNIX/Linux, page 52.*

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP3 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Enterprise Edition SP2 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Professional Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP Professional Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Business Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista Enterprise Edition SP1 (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP1 (on x64)</td>
</tr>
</tbody>
</table>

| Microsoft Excel | • Microsoft Excel 2010 |
|                | • Microsoft Excel 2007 SP2 |
|                | • Microsoft Excel 2003 SP3 |
|                | • Microsoft Excel 2002 (XP) SP3 |

**MicroStrategy product: Tutorial**

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>MicroStrategy</td>
<td>The MicroStrategy Tutorial project should only be used with the MicroStrategy 9.x product suite. This project can only be installed on Windows 32-bit and 64-bit operating systems.</td>
</tr>
<tr>
<td></td>
<td>See the <em>Introduction to MicroStrategy: Evaluation Guide</em> for complete details</td>
</tr>
</tbody>
</table>
## MicroStrategy product: Analytics Modules

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>MicroStrategy</td>
<td>It is highly recommended you use the MicroStrategy Analytics Modules only with the current release of MicroStrategy. This project can only be installed on Windows 32-bit and 64-bit operating systems.</td>
</tr>
</tbody>
</table>

## MicroStrategy product: Narrowcast Server

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td><strong>Certified:</strong> \n</td>
</tr>
<tr>
<td>Web browsers</td>
<td>Subscription Portal and Portal Administrator \n</td>
</tr>
</tbody>
</table>
## Planning Your Installation

### Installation prerequisites

**Third party Adobe Flash support**

MicroStrategy Narrowcast Server can deliver Adobe Flash content as the body of an email, as an email attachment, or to a file location. The support for Flash content deliveries by some third-party products is discussed below.

**Warning:** The third-party products discussed below are manufactured by vendors independent of MicroStrategy, and the information provided is subject to change. Refer to the appropriate third-party vendor documentation for updated Flash support information.

- Microsoft Outlook Express and Microsoft Outlook 2003 support Flash content both in the email body and as an attachment.
- Hotmail, YahooMail, and Gmail support Flash content only as a Flash.mht attachment. Flash content in the body of an email is not supported.
- Web browsers: Internet Explorer 7.x, Internet Explorer 8.x, Internet Explorer 9.x, Firefox 3.x, Firefox 4.x, Google Chrome 9.x, and Google Chrome 10.x support Flash content.

**Web server**

- Microsoft Internet Information Services 6
- Microsoft Internet Information Services 7
- Microsoft Internet Information Services 7.5

If you plan to use IIS 6 on 64-bit Windows operating systems, some MicroStrategy products must be configured as 64-bit applications, and some MicroStrategy products must be configured as 32-bit applications. These requirements are described in [Supporting MicroStrategy products with IIS 6 on 64-bit Windows operating systems, page 70](#).

- Microsoft Internet Information Services 7
- Microsoft Internet Information Services 7.5

**ODBC connectivity**

Connectivity to the Object Repository, Subscription Book Repository, and Portal Repository databases

**JDBC connectivity**

Sequelink (installed with the product)

**Export application**

Certified:

- Adobe Acrobat Reader 8.x, 9.x, and 10.x
- Microsoft Office 2002 (XP) SP3
- Microsoft Office 2003 SP3
- Microsoft Office 2007 SP2 (Microsoft Office 2007 .xlsx files can be included as a static attachment for a Narrowcast service. However, .xlsx files are not supported as imported templates in Narrowcast Server. Save the Excel 2007 .xlsx file in Excel 97-2003 workbook format. You can then import the file as a template.)
- Microsoft Office 2010
- Microsoft Office for Mac 2011

**Export devices**

Export devices can open a PDF file created by using MicroStrategy to export an object as a PDF. You can load a PDF file exported from MicroStrategy onto an export device in a couple of different ways. You can connect the export device directly to the machine that stores the PDF to load it onto the export device. You can also email the PDF file to the email account for the export device.

Certified:

- Nook firmware version 1.4
- Kindle firmware version 2.5.2

<table>
<thead>
<tr>
<th>Software</th>
<th>Software Requirements</th>
</tr>
</thead>
</table>
| Third party Adobe Flash | MicroStrategy Narrowcast Server can deliver Adobe Flash content as the body of an email, as an email attachment, or to a file location. The support for Flash content deliveries by some third-party products is discussed below. Warning: The third-party products discussed below are manufactured by vendors independent of MicroStrategy, and the information provided is subject to change. Refer to the appropriate third-party vendor documentation for updated Flash support information.  
- Microsoft Outlook Express and Microsoft Outlook 2003 support Flash content both in the email body and as an attachment.  
- Hotmail, YahooMail, and Gmail support Flash content only as a Flash.mht attachment. Flash content in the body of an email is not supported.  
- Web browsers: Internet Explorer 7.x, Internet Explorer 8.x, Internet Explorer 9.x, Firefox 3.x, Firefox 4.x, Google Chrome 9.x, and Google Chrome 10.x support Flash content. |
| Web server              | Microsoft Internet Information Services 6  
If you plan to use IIS 6 on 64-bit Windows operating systems, some MicroStrategy products must be configured as 64-bit applications, and some MicroStrategy products must be configured as 32-bit applications. These requirements are described in [Supporting MicroStrategy products with IIS 6 on 64-bit Windows operating systems, page 70](#).  
- Microsoft Internet Information Services 7  
- Microsoft Internet Information Services 7.5 |
| ODBC connectivity       | Connectivity to the Object Repository, Subscription Book Repository, and Portal Repository databases |
| JDBC connectivity       | Sequelink (installed with the product) |
| Export application      | Certified:  
- Adobe Acrobat Reader 8.x, 9.x, and 10.x  
- Microsoft Office 2002 (XP) SP3  
- Microsoft Office 2003 SP3  
- Microsoft Office 2007 SP2 (Microsoft Office 2007 .xlsx files can be included as a static attachment for a Narrowcast service. However, .xlsx files are not supported as imported templates in Narrowcast Server. Save the Excel 2007 .xlsx file in Excel 97-2003 workbook format. You can then import the file as a template.)  
- Microsoft Office 2010  
- Microsoft Office for Mac 2011 |
| Export devices          | Export devices can open a PDF file created by using MicroStrategy to export an object as a PDF. You can load a PDF file exported from MicroStrategy onto an export device in a couple of different ways. You can connect the export device directly to the machine that stores the PDF to load it onto the export device. You can also email the PDF file to the email account for the export device. Certified:  
- Nook firmware version 1.4  
- Kindle firmware version 2.5.2 |
**Intelligence Server Universal software requirements on UNIX/Linux**

For the exact information such as version numbers and space requirements, see the MicroStrategy readme file.

MicroStrategy Integrity Manager for UNIX and Linux platforms has the same requirements as Intelligence Server Universal. Therefore, you can use the information in this section for Intelligence Server Universal and Integrity Manager requirements on UNIX and Linux platforms.

The following MicroStrategy products require an X-windows-enabled environment on all UNIX and Linux platforms:

- GUI-based MicroStrategy Installation Wizard
- Diagnostics and Performance Logging tool
- Service Manager

The following requirements also apply to all UNIX and Linux platforms:

- A Web browser (for example, Firefox 3.6) is required for viewing readme, release notes, and online help.


The requirements listed below describe general requirements as well as requirements specific to the UNIX and Linux platforms:

- *Configuring shared memory resources, page 53*
- *Oracle Solaris, page 55*
- *IBM AIX, page 56*
- *HP-UX, page 57*
- *Red Hat and SUSE Linux, page 57*

Be aware of the following before reviewing the sections listed above:

- The operating systems listed are deemed supported or certified to reflect the level of internal testing that each configuration was
exposed to for the current release. MicroStrategy recommends using certified configurations over the supported configurations.

- MicroStrategy certifies and supports operating systems that are compatible with a set of CPU chipsets, referred to as CPU architectures, that are binary-compatible. MicroStrategy tests on at least one of the CPU chipsets within a set of binary-compatible CPU architectures for purposes of certifying and supporting operating systems with MicroStrategy products. A valid CPU architecture is provided in parentheses () to clarify the operating system software certified or supported for Intelligence Server Universal.

- All UNIX and Linux operating systems are 64-bit.

- For information on LDAP Servers certified and supported for LDAP authentication with various Intelligence Server machine environments, see the MicroStrategy Readme.

### Configuring shared memory resources

To improve the performance of MicroStrategy Intelligence Server Universal for large scale production applications, Intelligence Server Universal can be configured to use shared memory resources. To support this configuration, you must ensure that your Intelligence Server Universal host machine uses values greater than or equal to the resource limits described below.

During installation (on the System Requirements page, see *System Requirements, page 119*), you have the following options:

- **Exit the MicroStrategy 9 setup wizard to do the required system changes (Recommended):** Select this option to cancel the installation and make the required system resource limit changes to support shared memory resources. This option is recommended for production environments. Information on the recommended resource limits is provided below.

- **Allow the setup to reconfigure MicroStrategy 9 to use Pipe as the Default IPC Mechanism:** Select this option to disable the use of shared memory resources for Intelligence Server Universal, and instead use the pipe mechanism. Disabling the ability to use shared memory resources can potentially decrease the performance of your MicroStrategy applications, and therefore this is not recommended for production environments.
• **Keep Shared Memory as the Default IPC Mechanism. (MicroStrategy 9 may not work properly):** Select this option to keep your system resource limits set at their current values to support shared memory resources. While this allows you to continue installation with the current system resource limits, Intelligence Server may not function properly after installation. If you plan to use shared memory resources for enhanced performance of your production environments, you should select the first option to exit the installation and make the required system changes.

The tables below provide recommended values for various system resource limits on UNIX and Linux.

Modifying the system resource limits listed below can affect system-wide behavior and therefore, steps to modify these values are not given. You should refer to your UNIX and Linux documentation and contact your system administrator to modify these settings.

Shared memory settings on UNIX and Linux operating systems that may require modification to support Intelligence Server execution are listed in the table below:

<table>
<thead>
<tr>
<th>Setting Name</th>
<th>Description</th>
<th>Recommended Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>shmmni</td>
<td>Maximum number of shared memory identifiers at any given time.</td>
<td>2048</td>
</tr>
<tr>
<td>shmseg</td>
<td>Maximum number of segments per process.</td>
<td>2048</td>
</tr>
</tbody>
</table>

Note: This setting does not exist on Solaris, AIX, or Linux operating systems.

Semaphores are used to synchronize shared memory communications. The names of the settings that control semaphores differ between operating systems as listed in the tables below:

<table>
<thead>
<tr>
<th>Setting Name on Solaris, AIX, and HP-UX</th>
<th>Description</th>
<th>Recommended Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>semmni</td>
<td>Maximum number of system wide semaphore sets.</td>
<td>2048</td>
</tr>
<tr>
<td>semmns</td>
<td>Maximum number of semaphores in the system.</td>
<td>2048</td>
</tr>
</tbody>
</table>
### Oracle Solaris

<table>
<thead>
<tr>
<th>Setting Name on Linux</th>
<th>Description</th>
<th>Recommended Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>semmsl</td>
<td>Maximum number of semaphores in a semaphore set.</td>
<td>250</td>
</tr>
<tr>
<td>semmns</td>
<td>Maximum number of semaphores in the system.</td>
<td>32000</td>
</tr>
<tr>
<td>semopm</td>
<td>Maximum number of operations in a simple semop call.</td>
<td>32</td>
</tr>
<tr>
<td>semni</td>
<td>Maximum number of semaphore sets.</td>
<td>2048</td>
</tr>
</tbody>
</table>

**Operating System**

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Solaris 9.x</td>
<td>Oracle Solaris 9.x (on SPARC)</td>
</tr>
<tr>
<td></td>
<td><strong>Patch requirements:</strong> Oracle-recommended Patch Cluster dated 09/11/2008 (kernel patch 122300-31) or later.</td>
</tr>
<tr>
<td>Oracle Solaris 10.x</td>
<td>Oracle Solaris 10.x (on SPARC)</td>
</tr>
<tr>
<td></td>
<td><strong>Patch requirements:</strong> Oracle-recommended Patch Cluster dated 08/11/2009 (kernel patch 141414-08) or later.</td>
</tr>
<tr>
<td>Solaris Zones for Solaris 10.x</td>
<td>MicroStrategy supports the installation of Intelligence Server Universal on Solaris Zones which meet all applicable Solaris 10.x requirements. Refer to your third-party Oracle documentation for information on Solaris Zones.</td>
</tr>
</tbody>
</table>
## IBM AIX

<table>
<thead>
<tr>
<th>Operating Systems</th>
<th>Requirements</th>
</tr>
</thead>
</table>
| IBM AIX 5.3       | IBM AIX 5L Version 5.3 (on IBM POWER Architecture)  
Certified patch requirements: Technology Level 10, with the following filesets:  
- bos.rte.libc 5.3.0.00 or higher  
- XL C/C++ Runtime 9.0.0.0 or higher (xlC.rte)  
- C/C++ Runtime 9.0.0.0 or higher for AIX 5.3 (xlC.aix50.rte)  
- bos.perf.libperfstat 5.3.0.00 or higher  
Supported patch requirements: Technology Level 9, with the following filesets:  
- bos.rte.libc 5.3.0.00 or higher  
- XL C/C++ Runtime 9.0.0.0 or higher (xlC.rte)  
- C/C++ Runtime 9.0.0.0 or higher for AIX 5.3 (xlC.aix50.rte)  
- bos.perf.libperfstat 5.3.0.00 or higher  
Warning: To successfully register MicroStrategy Intelligence Server Universal as a service or support the use of CPU licenses on IBM AIX 5.3, you must obtain the IBM Authorized Program Analysis Report (APAR) IY81136. Contact IBM Production Support for help with obtaining and configuring this APAR. This APAR is available at the following URL, which is valid as of the release of this manual: http://www-01.ibm.com/support/docview.wss?rs=0&q1=IY81136&uid=isg1IY81136&loc=en_US&cs=utf-8&cc=us&lang=en |
| IBM AIX 6.1       | IBM AIX 6.1 (on IBM POWER Architecture)  
Patch requirements: Technology Level 1, with the following file sets:  
- bos.rte.libc 5.3.9.3 or higher  
- XL C/C++ Runtime V10.1.0.2 for AIX (xlC.aix50.rte)  
- bos.perf.libperfstat 5.3.9.1 or higher |
| LPAR for AIX      | MicroStrategy certifies the installation of Intelligence Server Universal on AIX 6.1 LPAR architecture for micropartitions, which meets all applicable AIX 6.1 requirements. This support also requires the use of MicroStrategy named user licenses.  
Refer to your third-party IBM documentation for information on LPAR.  
MicroStrategy supports the installation of Intelligence Server Universal on AIX LPAR architecture for dedicated partitions, which meets all applicable AIX requirements. |
### HP-UX

<table>
<thead>
<tr>
<th>Operating Systems</th>
<th>Requirements</th>
</tr>
</thead>
</table>
| HP-UX 11i v2      | HP-UX 11i v2 Foundation Operating Environment (on Intel Itanium)  
Patch requirements: Base Quality Pack Bundle for HP-UX 11i v2 dated December 2007  
HP-UX 11i v3 (on Intel Itanium)  
Patch requirements: Base Quality Pack Bundle for HP-UX 11i v3 dated September 2008 |

### Red Hat and SUSE Linux

<table>
<thead>
<tr>
<th>Operating Systems</th>
<th>Requirements</th>
</tr>
</thead>
</table>
| Red Hat Enterprise Linux 5.2, 5.3, 5.4, 5.5, and 5.6 | Red Hat Enterprise Linux 5.x is defined as support for versions 5.2, 5.3, 5.4, 5.5, and 5.6 for the following editions:  
• Red Hat Linux Enterprise Linux 5.x (on x86-64)  
• Red Hat Linux Enterprise Linux Advanced Platform 5.x (on x86-64)  
To support Red Hat Enterprise Linux 5.5 and 5.6, you must install pdksh-5.2.14-36.el5.x86_64.rpm, which is available on the Red Hat Enterprise Linux install media.  
To support the installation of Intelligence Server Universal on Red Hat Linux Enterprise Linux 5.x and Red Hat Linux Enterprise Linux Advanced Platform 5.x, you must install the krb5-libs-1.5-17.x86_64.rpm package.  
To support the use of a graphical user interface for MicroStrategy tools and installing MicroStrategy products on Red Hat Linux Enterprise Linux 5.x and Red Hat Linux Enterprise Linux Advanced Platform 5.x, you must install the libXp-1.0.0-8.x86_64.rpm package included with your Red Hat Enterprise Linux 5.x or Advanced Platform 5.x install media.  
To support MicroStrategy connections to databases on Red Hat Linux Enterprise Linux 5.x and Red Hat Linux Enterprise Linux Advanced Platform 5.x, you must install the krb5-libs-1.5-17.i386.rpm package.  
You must define SELinux as Permissive or Disabled. |
| Red Hat Enterprise Linux 6.0 | Red Hat Enterprise Linux 6.0 (on x86-64)  
You must install the packages compat-libstdc++-33-3.2.3-69.el6.i686.rpm and nss-pam-ldapd.i686.rpm, which are provided with the Red Hat Enterprise Linux install media. |
| SUSE Linux Enterprise Server 11 | SUSE Linux Enterprise Server 11.x (on x86-64) |
Supporting Intelligence Server Universal memory allocation on Linux

In addition to the operating system requirements listed above, MicroStrategy recommends that the Linux kernel setting \texttt{vm.max_map_count} be defined as 5,242,880 bytes. This allows Intelligence Server Universal to utilize system memory resources. If a lower value is used, Intelligence Server Universal may not be able to use all available system resources. This can cause some Intelligence Server Universal actions to fail due to lack of system resources, which could be completed if the additional system resources were made available by increasing the value for this kernel setting.

For information on this setting, including how to modify its value, refer to your third-party Linux operating system documentation.

MicroStrategy Web Universal software requirements and recommendations

To confirm the latest requirement information, see the MicroStrategy readme. For specific patches, filesets, technology level, and other requirements for UNIX and Linux operating systems, see \textit{Intelligence Server Universal software requirements on UNIX/Linux, page 52}. 

<table>
<thead>
<tr>
<th>Operating Systems</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Linux</td>
<td>Oracle Linux (on x86-64) distributed through the Oracle Unbreakable Linux program, for use with Red Hat Enterprise Linux. Refer to the MicroStrategy certification and support requirements for Red Hat Enterprise Linux for any additional requirements to support MicroStrategy systems on Oracle Linux.</td>
</tr>
</tbody>
</table>
## Web client software

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Operating system</strong></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 SP2 all editions (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 R2 SP2 all editions (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows XP SP3 all editions (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Vista SP1 or SP2 all editions (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 SP1 all editions (on x86 or x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 R2 SP1 all editions (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 SP2 all editions (on x64)</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Mac OS X</td>
</tr>
<tr>
<td></td>
<td>• Windows XP SP2 all editions (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 SP2 all editions (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 SP1 all editions (on x64)</td>
</tr>
<tr>
<td></td>
<td>• RedHat Enterprise Linux 5.2, 5.3, 5.4, 5.5, 5.6, and 6.0 (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• SUSE Linux Enterprise Server 10.x (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• SUSE Linux Enterprise Server 11.x (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• CentOS 5.3 (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• Ubuntu 8.0.4 (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• Ubuntu LTS 10.04 (on x86-64)</td>
</tr>
<tr>
<td><strong>Web browser (Windows client)</strong></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Microsoft Internet Explorer 7.x, 8.x, and 9.x</td>
</tr>
<tr>
<td></td>
<td>• Firefox 3.x and 4.x</td>
</tr>
<tr>
<td></td>
<td>• Google Chrome 9.x and 10.x</td>
</tr>
<tr>
<td><strong>Web browser (non-Windows client)</strong></td>
<td><strong>Supported Linux client:</strong></td>
</tr>
<tr>
<td></td>
<td>• Firefox 3.x and 4.x</td>
</tr>
<tr>
<td></td>
<td><strong>Supported Mac OS client:</strong></td>
</tr>
<tr>
<td></td>
<td>• Firefox 3.x and 4.x</td>
</tr>
<tr>
<td></td>
<td>• Safari 5.x</td>
</tr>
<tr>
<td></td>
<td>• Google Chrome 9.x and 10.x</td>
</tr>
<tr>
<td></td>
<td><strong>Supported (iPhone, iPod Touch, and iPad):</strong></td>
</tr>
<tr>
<td></td>
<td>• Safari web browser packaged with iOS 3.2.x (Web Reporter)</td>
</tr>
<tr>
<td></td>
<td>• Safari web browser packaged with iOS 4.2.x (Web Reporter)</td>
</tr>
<tr>
<td></td>
<td>• Safari web browser packaged with iOS 4.3.x (Web Reporter)</td>
</tr>
<tr>
<td><strong>Spreadsheet export application</strong></td>
<td><strong>Microsoft Excel or other spreadsheet application that supports the CSV file type:</strong></td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2010</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2007 SP2</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2003 SP3</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office 2002 (XP) SP3</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office for Mac 2011</td>
</tr>
<tr>
<td><strong>PDF viewer (for viewing and exporting PDF documents)</strong></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Adobe Acrobat Reader 8.x, 9.x, and 10.x</td>
</tr>
</tbody>
</table>
### Web server software

For information on the exact version numbers, see the MicroStrategy readme file.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
</table>
| Adobe Flash Player         | **Certified:**  
|                            | • Adobe Flash Player 10.2  
|                            | **Supported:**  
|                            | • Adobe Flash Player 10.1                                                                 |

#### Operating system

MicroStrategy Web Universal is certified or supported to work with the following operating systems.

**Supported:**
- Windows 2003 Standard Edition SP2 (on x86)
- Windows 2003 Enterprise Edition SP2 (on x86)
- Windows 2003 Standard Edition R2 SP2 (on x86)
- Windows 2003 Enterprise Edition R2 SP2 (on x86)

**Supported for demonstration purposes only:**
- Windows XP Professional Edition SP3 (on x86)
- Windows Vista Business Edition SP1 or SP2 (on x86)
- Windows 7 Professional Edition SP1 (on x86)
- Windows 7 Enterprise Edition SP1 (on x86)

MicroStrategy Web ASP.Net server on 64-bit operating systems

**Certified:**
- Windows Server 2008 Enterprise Edition SP2 (on x64)
- Windows Server 2008 Standard Edition SP2 (on x64)
- Windows Server 2008 Enterprise Edition R2 SP1 (on x64)
- Windows Server 2008 Standard Edition R2 SP1 (on x64)
- Windows 2003 Standard Edition SP2 (on x64)
- Windows 2003 Enterprise Edition SP2 (on x64)
- Windows 2003 Standard Edition R2 SP2 (on x64)
- Windows 2003 Enterprise Edition R2 SP2 (on x64)

**Supported:**
- Windows Server 2008 Enterprise Edition SP1 (on x64)
- Windows Server 2008 Standard Edition SP1 (on x64)
- Windows 7 Professional Edition SP1 (on x64), For demonstration purposes only
- Windows 7 Enterprise Edition SP1 (on x64), For demonstration purposes only
## Planning Your Installation

### Operating system (continued)

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Certified:</strong></td>
<td>MicroStrategy Web Universal JSP server on 64-bit operating systems</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 2003 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 2003 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 2003 Standard Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 2003 Enterprise Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• IBM AIX 5.3 Technology Level 10 (on IBM POWER Architecture)</td>
</tr>
<tr>
<td></td>
<td>• IBM AIX 6.1 Technology Level 1 (on IBM POWER Architecture)</td>
</tr>
<tr>
<td></td>
<td>• Oracle Solaris 9.x (on SPARC)</td>
</tr>
<tr>
<td></td>
<td>• Oracle Solaris 10.x (on SPARC)</td>
</tr>
<tr>
<td></td>
<td>• Hewlett Packard HP-UX 11i v2 (on Intel Itanium)</td>
</tr>
<tr>
<td></td>
<td>• Hewlett Packard HP-UX 11i v3 (on Intel Itanium)</td>
</tr>
<tr>
<td></td>
<td>• Red Hat Enterprise Linux 5.2, 5.3, 5.4, 5.5, and 5.6 and RedHat Enterprise Linux Advanced Platform 5.2, 5.3, 5.4, 5.5, and 5.6 (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• Red Hat Enterprise Linux 6.0 (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• SUSE Linux Enterprise Server 11.x (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• Oracle Linux (on x86-64) - distributed through Oracle Unbreakable Linux</td>
</tr>
<tr>
<td><strong>Supported:</strong></td>
<td>Windows 7 Professional Edition SP1 (on x64), for demonstration purposes only.</td>
</tr>
<tr>
<td></td>
<td>Windows 7 Enterprise Edition SP1 (on x64), for demonstration purposes only.</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2008 Enterprise Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>Windows Server 2008 Standard Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>IBM AIX 5.3 Technology Level 9 (on IBM POWER Architecture)</td>
</tr>
</tbody>
</table>

### Application servers

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MicroStrategy Web Universal</strong> is certified to work with the following application servers.</td>
<td></td>
</tr>
</tbody>
</table>

<p>| <strong>Certified:</strong> | MicroStrategy Web Universal JSP server general requirements |
|               | • Oracle Web Application Server 10g Release 3 (10.1.3) |
|               | • Oracle WebLogic 10.3 |
|               | • Oracle WebLogic 9.2 |
|               | • IBM WebSphere 6.1 |
|               | • IBM WebSphere 7.0 |
|               | • Apache Tomcat 6.0 |
|               | • Apache Tomcat 5.5 |
|               | • Sun Java System Application Server 9.1 |
|               | • JBoss Enterprise Application Platform 4.3 |
| <strong>Supported:</strong> | |
|             | • SAP NetWeaver Application Server 7.1 |
|             | • JBoss Application Server 4.2 |</p>
<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web servers</td>
<td>MicroStrategy Web Universal is certified to work with the following web servers.</td>
</tr>
<tr>
<td></td>
<td><strong>Certified</strong> MicroStrategy Web Universal JSP server general requirements:</td>
</tr>
<tr>
<td></td>
<td>• Apache 2.2</td>
</tr>
<tr>
<td></td>
<td>• Apache 2.0</td>
</tr>
<tr>
<td></td>
<td>• IBM HTTP Server 6.0.2</td>
</tr>
<tr>
<td></td>
<td>• IBM HTTP Server 6.1</td>
</tr>
<tr>
<td></td>
<td>• Oracle iPlanet Web Server 6.1</td>
</tr>
<tr>
<td></td>
<td>• Oracle iPlanet Web Server 7.0</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Internet Information Services 6.0 and 7.0</td>
</tr>
<tr>
<td></td>
<td>If you plan to use IIS 7 as the web server for MicroStrategy Web Universal, you must ensure</td>
</tr>
<tr>
<td></td>
<td>that some IIS options are enabled, as described in <em><a href="#">Supporting IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server, page 68</a></em>.</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Internet Information Services 7.5</td>
</tr>
<tr>
<td></td>
<td>If you plan to use IIS 7.5 as the web server for MicroStrategy Web Universal, you must ensure</td>
</tr>
<tr>
<td></td>
<td>that some IIS options are enabled, as described in <em><a href="#">Supporting IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server, page 68</a></em>.</td>
</tr>
<tr>
<td></td>
<td>Supported only for demonstration purposes on 32-bit operating systems</td>
</tr>
<tr>
<td></td>
<td><strong>Certified</strong> MicroStrategy Web ASP.Net server on 64-bit operating systems:</td>
</tr>
<tr>
<td></td>
<td>• Internet Information Services 6.0, 7.0, and 7.5</td>
</tr>
<tr>
<td></td>
<td>If you plan to use IIS 7 or 7.5 as the web server for MicroStrategy Web Universal, you must</td>
</tr>
<tr>
<td></td>
<td>ensure that some IIS options are enabled, as described in <em><a href="#">Supporting IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server, page 68</a></em>.</td>
</tr>
</tbody>
</table>
### Installation prerequisites

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portal server</td>
<td>MicroStrategy Portlets allow MicroStrategy Web Universal to be easily</td>
</tr>
<tr>
<td></td>
<td>integrated and configured on portal environments such as IBM WebSphere,</td>
</tr>
<tr>
<td></td>
<td>Microsoft SharePoint Portal, SAP Portal, and Oracle WebLogic.</td>
</tr>
<tr>
<td></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• IBM WebSphere Portal 6.1</td>
</tr>
<tr>
<td></td>
<td>• IBM WebSphere Portal 7.0</td>
</tr>
<tr>
<td></td>
<td>• Microsoft Office SharePoint Portal 2007</td>
</tr>
<tr>
<td></td>
<td>• Microsoft SharePoint Portal Server 2010</td>
</tr>
<tr>
<td></td>
<td>• Oracle WebLogic Portal 10.3.2</td>
</tr>
<tr>
<td></td>
<td>• Oracle WebLogic Portal 9.2</td>
</tr>
<tr>
<td></td>
<td>• SAP NetWeaver Composition Environment 7.1</td>
</tr>
<tr>
<td></td>
<td>• Liferay Portal 5.2</td>
</tr>
<tr>
<td></td>
<td>• Liferay Portal 6.0</td>
</tr>
<tr>
<td></td>
<td>• Drupal 6.20</td>
</tr>
<tr>
<td></td>
<td>• DotNetNuke 5.6 web content management system</td>
</tr>
<tr>
<td>GIS components</td>
<td>MicroStrategy GIS Connectors let you integrate with ESRI to create</td>
</tr>
<tr>
<td></td>
<td>sophisticated GIS applications. The following GIS components are required</td>
</tr>
<tr>
<td></td>
<td>to support this GIS integration.</td>
</tr>
<tr>
<td></td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• ArcGIS 9.3.1</td>
</tr>
<tr>
<td></td>
<td>• ArcGIS 10</td>
</tr>
</tbody>
</table>
### MicroStrategy Web Services J2EE software requirements and recommendations

To confirm the latest requirement information, see the MicroStrategy readme. For specific patches, filesets, technology level, and other
requirements for UNIX and Linux operating systems, see *Intelligence Server Universal software requirements on UNIX/Linux*, page 52.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td><strong>Certified:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Web Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP2 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition R2 SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Solaris 9.x (on SPARC)</td>
</tr>
<tr>
<td></td>
<td>• Solaris 10.x (on SPARC)</td>
</tr>
<tr>
<td></td>
<td>• AIX 5L Version 5.3 Technology Level 10 (on IBM POWER Architecture)</td>
</tr>
<tr>
<td></td>
<td>• AIX 6.1 Technology Level 1 (on IBM POWER Architecture)</td>
</tr>
<tr>
<td></td>
<td>• Red Hat Enterprise Linux 5.2, 5.3, 5.4, 5.5, and 5.6 and RedHat Enterprise Linux Advanced Platform 5.2, 5.3, 5.4, 5.5, and 5.6 (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• Red Hat Enterprise Linux 6.0 (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• Oracle Linux (on x86-64) distributed through the Oracle Unbreakable Linux program.</td>
</tr>
<tr>
<td></td>
<td>• SUSE Linux Enterprise Server 11.x (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• HP-UX 11i v2 Foundation Operating Environment (on Intel Itanium)</td>
</tr>
<tr>
<td></td>
<td>• HP-UX 11i v3</td>
</tr>
<tr>
<td></td>
<td><strong>Supported:</strong></td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Web Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Standard Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2003 Enterprise Edition R2 SP2 (on x86)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Enterprise Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows Server 2008 Standard Edition SP1 (on x64)</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Professional Edition SP1 (on x86 or x64), for demonstration purposes only</td>
</tr>
<tr>
<td></td>
<td>• Windows 7 Enterprise Edition SP1 (on x86 or x64), for demonstration purposes only</td>
</tr>
<tr>
<td></td>
<td>• SUSE Linux Enterprise Server 10 Patchlevel 2 (on x86-64)</td>
</tr>
<tr>
<td></td>
<td>• IBM AIX 5.3 Technology Level 9 (on IBM POWER Architecture)</td>
</tr>
</tbody>
</table>
### Requirement | Recommended
--- | ---
Application server | **Certified:**
• Oracle Web Application Server 10g Release 3 (10.1.3)
• Oracle WebLogic 10.3
• Oracle WebLogic 9.2
• IBM WebSphere 6.1
• IBM WebSphere 7
• Apache Tomcat 6.0
• Apache Tomcat 5.5
• Sun Java System Application Server 9.1
• JBoss Enterprise Application Platform 4.3
**Supported:**
• SAP NetWeaver Application Server 7.1
• JBoss Application Server 4.2

JDK, JRE, and JVM | • SUN JDK 1.5 (32-bit or 64-bit)
• IBM JDK 1.5 (32-bit or 64-bit)
• HP-UX JDK 1.5 (32-bit or 64-bit)
• Oracle JRockit Mission Control 3.1.0 for Java Version 6

---

**MicroStrategy Command Manager software requirements on UNIX/Linux**

For Command Manager operating system requirements on Windows platforms, see *MicroStrategy product: Command Manager, page 36*

### Software | Software Requirements
--- | ---
Operating system | **Certified:**
• Solaris 9.x (on SPARC)
• Solaris 10.x (on SPARC)
• AIX 5L Version 5.3 Technology Level 10 (on IBM POWER Architecture)
• AIX 6.1 Technology Level 1 (on IBM POWER Architecture)
• Red Hat Enterprise Linux 5.2, 5.3, 5.4, 5.5, and 5.6 and RedHat Enterprise Linux Advanced Platform 5.2, 5.3, 5.4, 5.5, and 5.6 (on x86-64)
• Red Hat Enterprise Linux 6.0 (on x86-64)
• Oracle Linux (on x86-64) distributed through the Oracle Unbreakable Linux program
• SUSE Linux Enterprise Server 11.x (on x86-64)
• HP-UX 11i v2 Foundation Operating Environment (on Intel Itanium)
• HP-UX 11i v3
**Supported:**
• IBM AIX 5.3 Technology Level 9 (on IBM POWER Architecture)
MicroStrategy Mobile software requirements for BlackBerry devices

The tables below list the MicroStrategy Mobile client application software requirements for BlackBerry, iPhone, iPod Touch, and iPad devices. To verify updated requirement information, see the MicroStrategy readme.

Software requirements for BlackBerry Devices

<table>
<thead>
<tr>
<th>BlackBerry Operating System</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlackBerry OS v4.x</td>
<td>Certified</td>
</tr>
<tr>
<td>BlackBerry OS v5.0</td>
<td>Supported</td>
</tr>
<tr>
<td>BlackBerry OS v6.0</td>
<td>Supported</td>
</tr>
</tbody>
</table>

Software requirements for iPhone and iPod Touch Devices

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS 3.2.x</td>
<td>Supported</td>
</tr>
<tr>
<td>iOS 4.2.x</td>
<td>Certified</td>
</tr>
<tr>
<td>iOS 4.3.x</td>
<td>Certified</td>
</tr>
</tbody>
</table>

Software requirements for iPad Devices

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS 3.2.x</td>
<td>Supported</td>
</tr>
<tr>
<td>iOS 4.2.x</td>
<td>Certified</td>
</tr>
<tr>
<td>iOS 4.3.x</td>
<td>Certified</td>
</tr>
</tbody>
</table>

MicroStrategy Mobile Server software requirements

The sections below list the MicroStrategy Mobile Server software requirements.
Mobile Server deployment requirements

- Mobile Server ASP.NET can be deployed using the same requirements listed for MicroStrategy Web (see *MicroStrategy product: MicroStrategy Web, page 45*).

- Mobile Server JSP can be deployed using the same requirements listed for MicroStrategy Web Universal (see *MicroStrategy Web Universal software requirements and recommendations, page 58*).

Web browsers for Mobile Server

The following web browsers are supported for Mobile Server.

<table>
<thead>
<tr>
<th>Web Browser</th>
<th>Certified</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Internet Explorer</td>
<td>Microsoft Internet Explorer 7.x</td>
</tr>
<tr>
<td></td>
<td>Microsoft Internet Explorer 8.x</td>
</tr>
<tr>
<td></td>
<td>Microsoft Internet Explorer 9.x</td>
</tr>
<tr>
<td>Firefox</td>
<td>Firefox 3.x</td>
</tr>
<tr>
<td></td>
<td>Firefox 4.x</td>
</tr>
<tr>
<td>Google Chrome</td>
<td>Google Chrome 9.x</td>
</tr>
<tr>
<td></td>
<td>Google Chrome 10.x</td>
</tr>
</tbody>
</table>

Supporting IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server

If you plan to use IIS 7 or IIS 7.5 as the web server for MicroStrategy Web or Mobile Server, you must ensure that some IIS options are enabled. The procedure below describes how to enable the options that are required to support IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server.

**To support IIS 7 as a web server for MicroStrategy Web or Mobile Server**

The third-party products discussed below are manufactured by vendors independent of MicroStrategy, and the steps to configure
these products is subject to change. Refer to the appropriate Microsoft
documentation for steps to configure IIS 7 or IIS 7.5.

1 On a Windows machine, open the Control Panel.

2 Double-click Programs and Features.

3 Click the Turn Windows features on or off task. The Windows Features
dialog box opens.

4 Expand Internet Information Services, and select the following options:
   a Expand Web Management Tools and select:
      • IIS Management Console
      • IIS Management Scripts and Tools
      • IIS Management Service
   b Expand World Wide Web Services, then expand Application
      Development Features, and select:
      • .NET Extensibility
      • ASP.NET
      • ISAPI Extensions
      • ISAPI Filters
   c Within World Wide Web Services, expand Common Http Features,
      and select:
      • Default Document
      • Static Content
   d Expand Security, and select:
      • Request Filtering
      • Windows Authentication

5 Click OK to save your changes.
Supporting MicroStrategy products with IIS 6 on 64-bit Windows operating systems

If you plan to use IIS 6 on 64-bit Windows operating systems, some MicroStrategy products must be configured as 64-bit applications, and some MicroStrategy products must be configured as 32-bit applications. These requirements are described below:

- For MicroStrategy products that must be configured as 32-bit applications on 64-bit Windows operating systems, you must define the IIS parameter `Enable32BitAppOnWin64` as True (1). These MicroStrategy products include:
  - Web MMT
  - Subscription Portal

- For MicroStrategy products that must be configured as 64-bit applications on 64-bit Windows operating systems, you must define the IIS parameter `Enable32BitAppOnWin64` as False (0). These MicroStrategy products include:
  - MicroStrategy Web Server (ASP.NET)
  - MicroStrategy Mobile Server (ASP.NET)
  - MicroStrategy Web Services (ASP.NET)

Be aware of the following:

- See your Microsoft documentation for steps to define the `Enable32BitAppOnWin64` parameter. Information on defining this parameter is available at [http://support.microsoft.com/kb/894435](http://support.microsoft.com/kb/894435).

- Since these MicroStrategy product groups require different IIS 6 settings, you can only install products from one of the product groups listed above on 64-bit Windows operating systems with IIS 6. For example, you cannot install and host MicroStrategy Mobile Server and Subscription Portal on the same 64-bit Windows operating system with IIS 6. If you use IIS 7, there are no restrictions as to the products that can be installed and hosted through IIS on 64-bit Windows operating systems.
Installation considerations

The following section contains guidelines and considerations you must keep in mind during MicroStrategy installation.

System sizing guidelines

The following topics describe sizing guidelines to consider when you initially set up MicroStrategy with your system. You should periodically reevaluate the system and update it based on actual system performance and use.

This section describes only the most basic guidelines. For detailed information refer to the MicroStrategy System Administration Guide.

Number of users

The number of users can be measured in the following ways:

- Total users: Users that are registered in the system. For example, if a corporate website is available to be viewed by 950 individuals, the site has 950 total users.

- Active users: Users that are logged into the system. If a site is available to be viewed by 950 total users and 30 of them are logged in to the site, there are 30 active users.

- Concurrent users: Users that have jobs being processed by a server (MicroStrategy Web, Intelligence Server, and so on) at the same time. For example, a site is available to 950 total users, and 30 people are logged in. Of 30 active users, 10 have jobs being processed by the server simultaneously, hence there are 10 concurrent users.

Of these measures, the number of concurrent users is important to consider. Your system must support the maximum number of concurrent users you expect at any given time.

Report complexity

The more complex a report, the more Intelligence Server resources are required. In this context, a “complex” report is one that requires a lot of
analytical processing. While reports with long, complicated SQL are certainly complex in nature, they do not necessarily require additional Intelligence Server resources to execute. It is the analytical processing in a report that creates additional stress on an Intelligence Server.

Since analytically complex reports create a heavier load on the Intelligence Server than simpler reports, you should have a general idea of what the average report complexity is for your system. Knowing this can help you decide on a caching strategy. For example, you may decide to pre-cache complex reports and determine the processing power your Intelligence Server needs.

The database server processes the SQL that Intelligence Server generates, so reports with extremely complex SQL can place additional stress on the database server. You should take this into account when sizing your database server machine.

### Ad hoc reports versus caches

Report *caches* store the results of previously executed reports. If a client (MicroStrategy Web, Desktop, and so on) requests a report that is cached, Intelligence Server simply returns the cached report results to the client. For any ad hoc reports that are not cached, Intelligence Server must go through the entire report execution cycle before it can return the results. For this reason, report caching allows better response time while minimizing the load on the Intelligence Server.

The benefits of caching are more apparent for complex reports than for simple reports. While caching a complex report may significantly improve execution time, a report cache for a simple report may not make much difference in this regard.

Therefore, the more complex the ad hoc reporting is in your system, the greater the overall load on the Intelligence Server. Be sure to take this into account when sizing your Intelligence Server machine.

The process for element browsing is similar to ad hoc reporting. *Element browsing* takes place when you navigate through hierarchies of attribute elements, for example, viewing the list of months in the year attribute. By default, caching is enabled for element browsing. In addition, you can limit the number of elements to be retrieved at a time.
Report Services document

Report Services documents utilize MicroStrategy objects to run complex and sophisticated reports. The datasets available to a document determine its content. Each dataset represents a report and its component objects, such as attributes, metrics, custom groups, and consolidations. When a dataset is available to a document, the entire report or any component object from that dataset can be included in the document output.

When creating a document, refer to the following guidelines to avoid an increase in the Intelligence Server execution time and the overall CPU usage:

- The datasets should be few in number, but large in size.
- The number of grids in the output document should be less in number. Consolidate the data to fit into fewer grids in the output document, where possible.
- Use of complex elements, such as consolidations, custom groups, and smart metrics can increase the Intelligence Server usage, especially if arithmetic operators are used in element definitions.
- Use Custom formatting only when required.

OLAP Services

OLAP Services store reports as Intelligent Cubes in the physical memory of the Intelligence Server. When these Intelligent Cubes are cached in memory, report manipulations, such as adding derived metrics and derived elements, formatting, and drilling within the Intelligent Cube, take considerably less time. This is the case because the new SQL is not run against the database.

OLAP Services provide enhanced report manipulation functionality at the cost of Intelligence Server resources, as the cubes are stored in the memory of the Intelligence Server. Consider the following factors to determine the size of the Intelligent Cubes:

- Intelligence Server resources
- Expected response time
- User concurrency

You must monitor Intelligence Server CPU utilization and memory usage closely as OLAP Services might have an impact on the performance of the platform, particularly the memory and report response time. For
information on OLAP Services, see the *OLAP Services Guide*. Additional performance tuning best practices for OLAP Services are provided in *Chapter 10, Managing Intelligent Cubes* of the *System Administration Guide*.

**Additional considerations**

Numerous factors can affect system performance, most of them related to system specifics, which makes them difficult to predict. Listed below are items you might want to consider when determining the requirements for your system:

- **Desktop versus MicroStrategy Web usage**—MicroStrategy products are designed with the assumption that the majority of users access the system through MicroStrategy Web while a smaller percentage use the Desktop products.

- **Statistics logging**—Statistics logging is very useful for analyzing and further refining the system configuration based on actual usage. However, logging all statistics all the time can create a noticeable increase in system response time. For this reason, you might choose to log only a subset of the statistics generated or only log statistics periodically.

- **Backup frequency**—Caches can be stored in memory and on disk. When you enable backup, you allow the Intelligence Server to write all cache files to disk. If the backup frequency is set to the default of zero, backup files are written to disk as soon as they are created. However, writing all cache files to disk all the time can cause a noticeable reduction in system performance.

  Set the backup frequency to a value that minimizes disk writes and optimizes memory usage for your system.

- **Ratio of MicroStrategy Web servers to Intelligence Servers**—In a typical system you should have a 1:1 ratio of Intelligence Servers to MicroStrategy Web servers. This ensures that resources on both sides are optimized. However, you might find it useful to add Intelligence Servers or MicroStrategy Web servers depending on your particular requirements.

- **Report Styles**—MicroStrategy Web provides a set of different XSL report styles. These styles provide an easy way for you to customize how reports look. Due to the varying complexity of these styles, some might require more processing than others.
MicroStrategy Professional Services for high performance

MicroStrategy Professional Services has identified five primary levers customers can use to get dramatically faster performance:

- Employ in-memory Business Intelligence
- Design high performance dashboards
- Optimize query efficiency
- Implement effective caching strategies
- Configure MicroStrategy for high performance

In just one week, MicroStrategy Professional Services, will conduct a thorough examination of your Business Intelligence implementation, providing you with actionable recommendations on these five key areas to improve overall performance. The MicroStrategy Performance Analysis service delivers:

- **Performance optimization roadmap**: A customized report with prioritized recommendations to achieve performance goals.
- **System configuration**: Optimum configuration setting recommendations to achieve efficient use of resources across different MicroStrategy products.
- **Performance monitoring plan**: A set of performance related metrics to proactively monitor and identify performance opportunities.


Common questions about sizing

The sections below provide brief explanations to common sizing questions. For detailed information on tuning your MicroStrategy environment, see *Chapter 7, Tuning your System for Best Performance* in the *System Administration Guide*. The sections below also provide other additional resources.
**Why should I increase the processor speed of Intelligence Server?**

Increasing the processor speed of Intelligence Server enhances performance and reduces execution time for all analytical tasks and for requests from the Extensible Markup Language (XML) and Component Object Model (COM) application programming interfaces (APIs). If you see that the machine or machines are running consistently at a high capacity, for example, greater than 80%, it may be a sign that a faster processor would improve the system’s capacity.

For more detailed information on tuning your processors for your MicroStrategy environment, see the section *Managing system resources* in the *System Administration Guide*.

**Why should I add more processors to Intelligence Server?**

Adding more processors to the Intelligence Server allows for a better load distribution among the processors. This provides an overall performance gain. If you notice that the processor is running consistently at a high capacity, for example, greater than 80%, consider increasing the number of processors.

For more detailed information on tuning your processors for your MicroStrategy environment, see the section *Managing system resources* in the *System Administration Guide*.

**Why should I increase memory on the machine that hosts Intelligence Server?**

If the physical disk is utilized too much on a machine hosting Intelligence Server, it can indicate that there is a bottleneck in the system’s performance. To monitor this on a Windows machine, use the Windows Performance Monitor for the object **PhysicalDisk** and the counter **% Disk Time**. If you see that the counter is greater than 80% on average, it may indicate that there is not enough memory on the machine.

For more detailed information on tuning your machine’s memory for your MicroStrategy environment, see the section *Managing system resources* in the *System Administration Guide*.

**What would more network bandwidth do for me?**

You can tell whether or not your network is negatively impacting your system’s performance by monitoring how much of your network’s capacity is
being used. To monitor this on a Windows machine, use the Windows Performance Monitor for the object **Network Interface**, and the watch the counter **Total bytes/sec** as a percent of your network’s bandwidth. If it is consistently greater than 60% (for example), it may indicate that the network is negatively affecting the system’s performance.

For very large result sets, increasing network bandwidth reduces bottlenecks created by network congestion. The result is larger data flow and faster query response time.

For more detailed information on tuning your network for your MicroStrategy environment, see the section *How the network can affect performance* in the *System Administration Guide*.

**How many CPUs can a user fully utilize?**

One user can fully utilize up to one CPU, regardless of the number of CPUs available in the server. The load is split across multiple CPUs in multi-processor servers.

For more detailed information on how licensing can affect the utilization of CPUs, see *Chapter 4, Managing Your Licenses* in the *System Administration Guide*.

**What is the advantage of using hyper-threading for a dual processor?**

The advantage of using hyper-threading with a dual processor is that it decreases the overall CPU usage. The use of hyper-threading is recommended if you have a large number of users.

**What is the disadvantage of using hyper-threading for a dual processor?**

The disadvantage of using hyper-threading is that it increases the Intelligence Server execution time slightly. Therefore, for faster processing, the use of hyper-threading is not recommended.

**What is the largest Intelligent Cube size that I can store in an Intelligence Server?**

Intelligent Cubes must be stored in Intelligence Server memory for reports to access their data. While this can improve performance of these reports, loading too much data onto Intelligence Server memory can have a negative
impact on Intelligence Server’s ability to process jobs. For this reason, it is important to govern how much Intelligent Cube data can be stored on the Intelligence Server.

For information on governing Intelligent Cube memory usage, loading, and storage, see Chapter 10, Managing Intelligent Cubes in the System Administration Guide.

International support

The following table lists the language selection possibilities for different installation cases.

<table>
<thead>
<tr>
<th>Installation</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fresh installation on a system in which MicroStrategy application has never been installed before</td>
<td>The MicroStrategy Installation Wizard prompts you to select the language from the drop-down list. The user language in the product interface is the language that you select during installation. Once the product is installed, the Online Help is displayed in the same language that the user selects in the language prompt of the installation routine.</td>
</tr>
<tr>
<td>Repair or maintenance installation on a system on which MicroStrategy application has been installed before</td>
<td>All subsequent executions of the installation routine are displayed in the language that you selected the first time you installed the product on the system. The user language in the product interface is also the language that you selected the first time you installed the product on the system.</td>
</tr>
<tr>
<td>Completely uninstalling all the MicroStrategy products and installing the same version or a newer version</td>
<td>If you uninstall all the products and install either the same version or a higher version again, the MicroStrategy Installation Wizard prompts you to select the language from the drop-down list. <strong>Note:</strong> Even if you select a language from the language prompt in the installation routine, it has no effect on the default language of the product interfaces.</td>
</tr>
</tbody>
</table>

During installation, the installation Online Help is displayed in English only.

During installation, the installation Online Help is displayed in English only.
MicroStrategy Web and Intelligence Server compatibility

You must ensure the versions of MicroStrategy Web and Intelligence Server are compatible. For example, MicroStrategy Web 8.0.2 can only connect to Intelligence Server 8.0.0 or later. For a complete list of compatible MicroStrategy Web and Intelligence Server versions, refer to the MicroStrategy readme.

Refer to the MicroStrategy 9 General Information Readme for the complete MicroStrategy platform compatibility and interoperability specification. In addition, you can contact MicroStrategy Technical Support for the latest information and updates.

Certified ODBC drivers for MicroStrategy Intelligence Server

The following table identifies the available ODBC drivers for Windows, Solaris, AIX, HP-UX, and Redhat or SUSE Linux that are certified for Intelligence Server and different DBMS types. MicroStrategy-branded ODBC drivers are installed with the MicroStrategy products.

Be aware of the following when reviewing the table below:

- The following abbreviations mark certified configurations in the table below:

  **WH**: Data warehouse.

  **MD**: Metadata repository. MD also signifies certification or support for History List repositories, except where explicitly noted otherwise.

  **ST**: Statistics repository.

  **NCS**: MicroStrategy Narrowcast Server.

- MicroStrategy certifies 32-bit ODBC drivers for connection to DBMSs on Windows, Solaris, AIX, and Linux. MicroStrategy does not certify 64-bit ODBC drivers for DBMSs on these operating systems.

- MicroStrategy certifies 64-bit ODBC drivers for connection to DBMSs on the HP-UX platform.

- Only certified configurations are listed in the table below. For a complete list of certified and supported configurations with exact
version numbers, refer to the certified and supported configurations listed in the MicroStrategy General Information readme.

<table>
<thead>
<tr>
<th>DBMS</th>
<th>Driver</th>
<th>32-bit ODBC drivers</th>
<th>64-bit ODBC drivers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Windows</td>
<td>Solaris</td>
</tr>
<tr>
<td>Aster nCluster:</td>
<td>nCluster ODBC Driver</td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>• 4.5.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Greenplum:</td>
<td>MicroStrategy ODBC Driver for Greenplum</td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>• 3.2.x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 3.3.x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 4.0.x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 4.1.x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hadoop Hive 0.7</td>
<td>Hive ODBC Driver from Cloudera</td>
<td>WH</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hive ODBC Driver with HIVE 1101 patch (open source)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HP Neoview:</td>
<td>ODBC Driver for HP Neoview</td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>• 2.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 2.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBM DB2 for UDB:</td>
<td>MicroStrategy ODBC Driver for DB2 Wire Protocol</td>
<td>MD</td>
<td>WH</td>
</tr>
<tr>
<td>• 9.5 with Fix pack 7</td>
<td></td>
<td>ST</td>
<td>ST</td>
</tr>
<tr>
<td>• 9.7 Fix pack 3a</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBM DB2 for i:</td>
<td>MicroStrategy ODBC Driver for DB2 Wire Protocol</td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>• 6.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 7.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 3.4.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 3.5.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 11.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Informix Ultimate Edition:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 11.7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Informix XPS 8.5</td>
<td>MicroStrategy ODBC Driver for Informix 8 with Informix Client SDK</td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>Kognitio WX2 7.1.x</td>
<td>ODBC Driver for Kognitio WX2</td>
<td>WH</td>
<td></td>
</tr>
<tr>
<td>DBMS</td>
<td>Driver</td>
<td>32-bit ODBC drivers</td>
<td>64-bit ODBC drivers</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>---------------------------------------------</td>
<td>---------------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>Microsoft SQL Server:</td>
<td>MicroStrategy ODBC Driver for SQL Server Wire Protocol</td>
<td>MD WH ST</td>
<td>MD WH ST</td>
</tr>
<tr>
<td>• 2000 SP3/SP3a/SP4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 2005 SP1/SP2/SP3/SP4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 2008 SP1/2008 SP2/2008 R2</td>
<td>MS SQL Server</td>
<td>MD WH ST NCS</td>
<td></td>
</tr>
<tr>
<td>Microsoft SQL Server Parallel Data Warehouse 2008 R2</td>
<td>Microsoft SQL Server 2008 R2 Parallel Data Warehouse ODBC Client</td>
<td>WH</td>
<td></td>
</tr>
<tr>
<td>MySQL Community Server 5.5.8</td>
<td>MySQL ODBC Driver</td>
<td>MD WH</td>
<td>MD WH</td>
</tr>
<tr>
<td>MySQL Enterprise 5.5.8</td>
<td>MicroStrategy ODBC Driver for MySQL Wire Protocol</td>
<td>MD WH</td>
<td>MD WH</td>
</tr>
<tr>
<td>Netezza:</td>
<td>Netezza ODBC Driver</td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>• 5.0.x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 6.0.x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oracle:</td>
<td>MicroStrategy ODBC Driver for Oracle Wire Protocol</td>
<td>MD WH ST NCS</td>
<td></td>
</tr>
<tr>
<td>• 10g</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 10gR2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 11g</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 11gR2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ParAccel 3.0</td>
<td>ODBC Driver for ParAccel</td>
<td>WH</td>
<td></td>
</tr>
<tr>
<td>PostgreSQL:</td>
<td>MicroStrategy ODBC Driver for PostgreSQL Wire Protocol</td>
<td>MD WH</td>
<td>MD WH</td>
</tr>
<tr>
<td>• 8.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 9.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Brick 6.3</td>
<td>Red Brick Driver</td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>SAND CDBMS 6.1</td>
<td>Nucleus ODBC Driver</td>
<td>WH</td>
<td></td>
</tr>
<tr>
<td>Sybase ASE:</td>
<td>MicroStrategy ODBC Driver for Sybase ASE Wire Protocol</td>
<td>MD WH ST</td>
<td>MD WH ST</td>
</tr>
<tr>
<td>• 15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 15.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sybase I:</td>
<td>Sybase IQ ODBC Driver</td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>• 15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 15.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 15.2</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

© 2011 MicroStrategy, Inc.
The MicroStrategy Connectivity Wizard exposes only the MicroStrategy-branded ODBC drivers. However, this guide also provides instructions for drivers from other vendors, that you can install separately and use with MicroStrategy. For more information, refer to Appendix A, Connecting to Databases: ODBC and DSNs.

### Recommended system settings for UNIX and Linux

UNIX and Linux systems allow processes and applications to run in a virtual environment. This means that each process, depending on its owner and the settings for certain environment variables, are run using a distinct set of properties that affect how much memory the process can use, how many CPU seconds it can use, what thread model it can use, how many files it can open, and so on.

MicroStrategy Intelligence Server Universal installs on UNIX and Linux systems with the required environment variables set to ensure that the server's jobs are processed correctly. However, as mentioned above, some settings are related to the user who starts the process (also known as the owner of the process) and other settings can only be set by the system administrator. Some of these settings may also have limits enforced for reasons unrelated to supporting MicroStrategy.

The table below lists MicroStrategy’s recommendations for system settings that can affect the behavior of Intelligence Server Universal.

Modifying the system settings listed below can affect system-wide behavior and therefore, steps to modify these values are not given.

<table>
<thead>
<tr>
<th>DBMS</th>
<th>Driver</th>
<th>32-bit ODBC drivers</th>
<th>64-bit ODBC drivers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Windows</td>
<td>Solaris</td>
</tr>
<tr>
<td>Teradata:</td>
<td>Teradata ODBC Driver</td>
<td>MD</td>
<td>MD</td>
</tr>
<tr>
<td>• V12</td>
<td></td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>• V13</td>
<td></td>
<td>ST</td>
<td>ST</td>
</tr>
<tr>
<td>• V13.10</td>
<td></td>
<td>NCS</td>
<td>NCS</td>
</tr>
<tr>
<td>Vertica</td>
<td>Vertica ODBC driver</td>
<td>WH</td>
<td>WH</td>
</tr>
<tr>
<td>• 4.0.x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• 4.1.x</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Teradata:
- V12
- V13
- V13.10

Vertica:
- 4.0.x
- 4.1.x
You should refer to your UNIX and Linux documentation and contact your system administrator to modify these settings.

<table>
<thead>
<tr>
<th>Setting Name</th>
<th>Description</th>
<th>Recommended Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>cputime (time)</td>
<td>Maximum CPU seconds per process.</td>
<td>Unlimited.</td>
</tr>
<tr>
<td>filesize (file)</td>
<td>Maximum size for a single file.</td>
<td>Unlimited, or as large as the file system allows. Your system administrator may enforce limits on the maximum size of files for reasons unrelated to MicroStrategy. This value must be at least as large as the maximum size for core dump files (coredumpsize). <strong>Warning:</strong> AIX machines use a default filesize limit of 2 GB. To support copying the MicroStrategy installation file to an AIX machine, you may need to increase the filesize limit to be larger than the size of the MicroStrategy installation file.</td>
</tr>
<tr>
<td>datasize (data)</td>
<td>Maximum heap size per process.</td>
<td>Unlimited, or as large as the system virtual memory allows. Your system’s virtual memory constraints affect the data size you can set for a process’s heap size. The value should be the same as the maximum size for core dump files (coredumpsize).</td>
</tr>
<tr>
<td>stacksize (stack)</td>
<td>Maximum stack size per process.</td>
<td>200 MB.</td>
</tr>
<tr>
<td>coredumpsize (coredump)</td>
<td>Maximum size for a single core dump file.</td>
<td>Set this value to the same value as the maximum heap size per process (datasize). If core dump files are created that are larger than this value, the files are corrupted and unusable.</td>
</tr>
<tr>
<td>memoryuse (memory)</td>
<td>Maximum size of physical memory allotted per process.</td>
<td>Unlimited, or as large as the physical memory of your system allows.</td>
</tr>
<tr>
<td>vmemoryuse (vmemory)</td>
<td>Maximum size of virtual memory allowed per process.</td>
<td>Unlimited, or as large as your system virtual memory allows.</td>
</tr>
<tr>
<td>descriptors (nofiles)</td>
<td>Maximum number of file descriptors (open files) per process.</td>
<td>8192.</td>
</tr>
</tbody>
</table>
Methods of installation

The methods of MicroStrategy installation are:

- *Graphical user interface* (GUI)
- *Command line*
- *Silent installation*

Graphical user interface

The GUI mode presents a user interface for each page in the MicroStrategy Installation Wizard. You click the mouse to place the cursor on the desired object, then proceed as appropriate to complete the task. The following navigational buttons are also displayed:

- **Next**: Click to proceed to the next page.
- **Back**: Click to return to the previous page.
- **Cancel**: Click to cancel the installation and close the MicroStrategy Installation Wizard.
- **Finish** (only on the MicroStrategy Installation Wizard Complete page): Click to complete the setup and close the wizard.

MicroStrategy provides the following types of GUI installation:

- Typical
- Advanced

For information on how to perform these installations, refer to *Chapter 2, Installing MicroStrategy on Windows* and *Chapter 3, Installing MicroStrategy on UNIX and Linux*.

Typical installation

This installation is interactive and prompts you for a License Key, the products to be installed, the location for the MicroStrategy home directory, and the name of the program folder where the application shortcuts reside.
Advanced installation

This is similar to the Typical installation, except that you have more control over the individual directories in which each component is installed. The Advanced installation prompts you for the target directory for each component being installed.

Command line

In command line mode, you type the appropriate information at the prompt and press ENTER. Instructions are included on each page of the MicroStrategy Installation Wizard.

In some cases, you are asked to make a selection by pressing 1 or 2, and ENTER. You then press 0 and ENTER to continue.

Defaults appear next to each prompt and are enclosed in square brackets, for example, [1]. Press ENTER to use the default, or type a different response to the prompt to override the default.

In addition, on the command line wizard pages, the following options are available:

- Press 1 and then press ENTER to proceed to the next page.
- Press 2 and then press ENTER to return to the previous page.
- Press 3 and then press ENTER to cancel the installation and close the MicroStrategy Installation Wizard.
- On the last page, which is MicroStrategy Installation Wizard Complete, press 3 and then press ENTER to complete the setup and close the wizard.

For information on command line installation, refer to Chapter 3, Installing MicroStrategy on UNIX and Linux.

Silent installation

A silent, or unattended, installation is one that presents no graphical user interface (GUI). Silent installations are useful for system administrators who do not want users to run the installation themselves. It allows you to automate the installation, so it can be called from a script and executed without user interaction.
For information on silent installation, refer to Silent installation, page 398, in Chapter 9, Automated Installation on Windows and Silent installation, page 406, in Chapter 10, Automated Installation on UNIX and Linux.

Licensing information

If you have installed the Evaluation version of MicroStrategy, you cannot use its license key with a Generally Available (GA) license key in the same environment. Hence, the Evaluation version of MicroStrategy cannot be used for your production environment.

For example, if you have MicroStrategy 7.5.0 installed, your installation does not support MicroStrategy Office. This is due to the fact that its license key does not contain the license for MicroStrategy Office. If you want to test the capabilities of MicroStrategy Office, you can run the evaluation license key on your client computers. This can be done after downloading MicroStrategy Office, to test it in your environment. You must then configure MicroStrategy Office to connect to your development Intelligence Server, which was installed by the permanent GA license key, on the MicroStrategy Server computer. However, after attempting to run MicroStrategy Office, the following error message appears:

The selected Login ID does not have privilege to use MicroStrategy Office.

This message appears because the license keys on the client machine and on the Intelligence Server are different. Therefore, the same license key must be used on both Intelligence Server and on the client machine for the client to connect to Intelligence Server.

A workaround for this scenario is to create a testing environment, built entirely on the Evaluation license key. This enables customers to test any products they want to evaluate.

Types of licenses

Refer to your MicroStrategy contract and any accompanying contract documentation for descriptions of the different MicroStrategy license types.
Installation and configuration checklists

This guide provides information on how to install and configure MicroStrategy products on Windows, UNIX, and Linux. To help you navigate through this guide, the following sections in this chapter list the chapters that you should refer to depending on the platform on which you are installing MicroStrategy products. Each list also provides a brief overview of each chapter. It is recommended that you read this section before performing an installation. You can use the tables as checklists of installation and configuration tasks to be completed.

The appendixes in this guide are not listed in the checklists below. The checklists only cover the main steps to install and configure MicroStrategy products. The appendixes in this guide contain important configuration details that are useful throughout the life cycle of your MicroStrategy installation.

Installing and configuring MicroStrategy on Windows

If you are installing MicroStrategy on Windows, you should refer to the following chapters sequentially.

You can use the Complete column on the left to check off each high-level step as you complete it.

<table>
<thead>
<tr>
<th>Complete</th>
<th>Chapter and Installation Task</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Chapter 1, Planning Your Installation</strong>: Review this chapter for important installation prerequisites and considerations.</td>
</tr>
<tr>
<td></td>
<td><strong>Chapter 2, Installing MicroStrategy on Windows</strong>: This chapter describes the procedures for installing the MicroStrategy products necessary to run your business intelligence application in a Windows environment. Or <strong>Chapter 9, Automated Installation on Windows</strong>: As an alternative to the regular installation, you can perform a fully automated and unattended installation including customization routines available with the product. This chapter describes different types of unattended and automated installations and provides steps to perform these installations on Windows. Additionally, <strong>Chapter 11, Deploying OEM Applications</strong> explains the common workflow for deploying the MicroStrategy platform as an Original Equipment Manufacturer (OEM) application.</td>
</tr>
<tr>
<td></td>
<td><strong>Chapter 4, Activating Your Installation</strong>: After installing MicroStrategy products, you have 30 days to activate your software installation. If you have not activated your software after these 30 days have passed, some MicroStrategy features may become unavailable until you complete the software activation.</td>
</tr>
</tbody>
</table>
Chapter 5, Configuring and Connecting Intelligence Server: After installing and activating MicroStrategy products, you must use the MicroStrategy Configuration Wizard to configure the MicroStrategy metadata repository, statistics tables, history list tables, Intelligence Server, and project sources. This chapter describes the steps used to configure an installed MicroStrategy suite of products using the MicroStrategy Configuration Wizard.

Chapter 6, Deploying MicroStrategy Web and Web Universal: You can deploy your project to your user community using MicroStrategy Web or Web Universal. This chapter provides information on how to deploy and configure MicroStrategy Web or Web Universal on Windows, UNIX, and Linux platforms with various Web and application servers.

You can deploy your project with either MicroStrategy Web or Web Universal. MicroStrategy Web Universal is platform-independent, whereas MicroStrategy Web can be installed only on Windows.

Chapter 7, Deploying MicroStrategy Web Services: This chapter describes the procedure to deploy MicroStrategy Web Services for ASP.NET and J2EE platforms. MicroStrategy Web Services is used to support MicroStrategy Office.

Chapter 13, Adding or Removing MicroStrategy Components: This chapter describes the steps to add and remove MicroStrategy components on Windows, as well as other operating systems. For Windows platforms, refer to the following sections:

- Adding or removing MicroStrategy components on Windows.
- Re-installing MicroStrategy components on Windows.
- Uninstalling MicroStrategy components on Windows.
Installing and configuring MicroStrategy on UNIX and Linux

If you are installing MicroStrategy on UNIX and Linux operating systems, you should refer to the following chapters sequentially.

You can use the Complete column on the left to check off each high-level step as you complete it.

<table>
<thead>
<tr>
<th>Complete</th>
<th>Chapter and Installation Task</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><em>Chapter 1, Planning Your Installation:</em> Review this chapter for important installation prerequisites and considerations.</td>
</tr>
<tr>
<td></td>
<td><em>Chapter 3, Installing MicroStrategy on UNIX and Linux:</em> This chapter describes the procedures for installing the MicroStrategy products necessary to run your business intelligence application on a UNIX and Linux environment. Or <em>Chapter 10, Automated Installation on UNIX and Linux:</em> As an alternative, you can perform a fully automated and unattended installation without using the graphical user interface. This chapter describes different types of unattended and automated installations and steps to perform these installations on UNIX. Additionally, <em>Chapter 11, Deploying OEM Applications</em> explains the common workflow for deploying the MicroStrategy platform as an Original Equipment Manufacturer (OEM) application.</td>
</tr>
<tr>
<td></td>
<td><em>Chapter 4, Activating Your Installation:</em> After installing MicroStrategy products, you have 30 days to activate your software installation. If you have not activated your software after these 30 days have passed, some MicroStrategy features may become unavailable until you complete the software activation.</td>
</tr>
<tr>
<td></td>
<td><em>Chapter 5, Configuring and Connecting Intelligence Server:</em> After installing and activating MicroStrategy products, you must use the MicroStrategy Configuration Wizard to configure the MicroStrategy metadata repository, statistics tables, history list tables, Intelligence Server, and project sources. This chapter addresses the processes necessary to configure an installed MicroStrategy suite of products using the Configuration Wizard. If no project sources are defined, then the Configuration Wizard opens. Or <em>Chapter 12, Configuring MicroStrategy Using Command Line Tools:</em> MicroStrategy tools are provided in command line mode on UNIX and Linux so that you can perform various configuration tasks through the operating system console. This enables you to perform your required configurations even if you do not have access to the MicroStrategy interface.</td>
</tr>
<tr>
<td></td>
<td><em>Chapter 6, Deploying MicroStrategy Web and Web Universal:</em> You can deploy your project to your user community using MicroStrategy Web Universal. This chapter provides information on how to deploy and configure MicroStrategy Web and Web Universal on Windows, UNIX, and Linux platforms with various Web and application servers. You can deploy your project with either MicroStrategy Web or Web Universal. MicroStrategy Web Universal is platform-independent, whereas MicroStrategy Web can be installed only on Windows.</td>
</tr>
<tr>
<td>Complete</td>
<td>Chapter and Installation Task</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td><strong>Chapter 7, Deploying MicroStrategy Web Services:</strong> This chapter describes the procedure to deploy MicroStrategy Web Services for ASP.NET and J2EE platforms. MicroStrategy Web Services provides a standard SOAP-based implementation of XML Web Services. It is used by MicroStrategy Office and can also support MicroStrategy SDK programming.</td>
<td></td>
</tr>
<tr>
<td><strong>Chapter 8, Setting Up Documents and HTML Documents:</strong> This chapter explains the setup required for the Intelligence Server to create and execute HTML documents and documents. It also describes the steps to create this setup, which are only necessary on a UNIX or Linux environment.</td>
<td></td>
</tr>
<tr>
<td><strong>Chapter 13, Adding or Removing MicroStrategy Components:</strong> This chapter describes the steps to add and remove MicroStrategy components on all supported operating systems. For UNIX and Linux platforms, refer to the following section:</td>
<td></td>
</tr>
<tr>
<td>• <strong>Uninstalling MicroStrategy components on UNIX and Linux</strong></td>
<td></td>
</tr>
</tbody>
</table>
Introduction

This chapter describes the procedures for installing the MicroStrategy products that are necessary to run your business intelligence application on a Windows environment.

Before installing MicroStrategy products, you should refer to Chapter 1, Planning Your Installation for important pre-installation information.

Some MicroStrategy products are available in two versions, as described below.

- **Windows only**: The Windows only versions, labeled as MicroStrategy Intelligence Server, MicroStrategy Web, and so on, are compatible only with a Windows platform. With these versions, MicroStrategy Web can be deployed quickly and easily using MicroStrategy’s Internet Information Services (IIS) Web Server. The drawback is that IIS is the only Web server that can be used to deploy the Windows only version of MicroStrategy Web.

  The Windows only version of Intelligence Server can be used with MicroStrategy Web Universal.
• **Universal (platform independent):** The universal versions, labeled as MicroStrategy Intelligence Server Universal, MicroStrategy Web Universal, and so on, are compatible with Windows as well as UNIX/Linux platforms. Installing the universal versions on Windows lets you deploy MicroStrategy Web Universal with different application and Web server combinations. For example, instead of using IIS to deploy MicroStrategy Web Universal, you can use Apache Tomcat, Oracle 10g, and so on.

If you have used the Evaluation Edition of the MicroStrategy platform, you may have installed most of these products already. However, additional considerations are important when you are setting up a production business intelligence system as opposed to running the evaluation software. You should read this chapter carefully, even if you already have a working system from your Evaluation Edition.

This chapter has the following sections:

- *Installation procedure, page 92*
- *Configuring your MicroStrategy installation, page 109*

If you are installing MicroStrategy on UNIX or Linux, refer to *Installation procedures on UNIX and Linux, page 112* in Chapter 3, *Installing MicroStrategy on UNIX and Linux*.

Additionally, *Chapter 11, Deploying OEM Applications* explains the common workflow for deploying the MicroStrategy platform as an Original Equipment Manufacturer (OEM) application.

## Installation procedure

The MicroStrategy Installation Wizard provides steps to guide you through installing one or more MicroStrategy products in a Windows environment. The following sections can assist you in installing MicroStrategy products:

- *Installing with the MicroStrategy Installation Wizard, page 93*
- *Installation verification, page 108*

There are a number of installation alternatives and procedures to support your MicroStrategy installation documented in this guide, including the following:
• For information on installation prerequisites, see *Installation prerequisites, page 19* in *Chapter 1, Planning Your Installation*.

• For information about advanced installation functionality, such as installing in an SMS environment or using installation response files, see *Chapter 9, Automated Installation on Windows*.

• For information about installing and deploying MicroStrategy Web Universal with other Web and application servers, see *Chapter 6, Deploying MicroStrategy Web and Web Universal*.

• For information about deploying MicroStrategy Web Services ASP.NET and J2EE, see *Chapter 7, Deploying MicroStrategy Web Services*.

Note the following:

• If you have not uninstalled previous versions of MicroStrategy products, you are prompted to overwrite them. Click **Yes** to ensure that all products are installed properly. To retain the existing Tutorial metadata repository and warehouse, rename it or move it to another location before you start the installation process.

• Although MicroStrategy supports Windows Terminal Services, using Windows Terminal Services is not recommended for installation. It can affect the functionality of some MicroStrategy components.

**Installing with the MicroStrategy Installation Wizard**

To install MicroStrategy products, you must log on to your machine using a domain account with Windows administrative privileges for the domain or target machine. The domain must include your database servers.

To exit the installation process at any time, click **Cancel**.

---

**To access the MicroStrategy Installation Wizard**

1. Log on to the machine where you are installing one or more MicroStrategy products.

2. Exit all Windows applications before beginning the installation process.
You can access the MicroStrategy Installation Wizard in the following ways:

- **From the disk:** Insert the disk into the disk drive and wait for the MicroStrategy Main Menu window to display automatically.

  If the MicroStrategy Main Menu does not display, locate and run `Setup.exe` on the disk.

- **From the download site:** Download the files from the MicroStrategy download site. Locate and run the `Setup.exe` file.

Note the following:

- Contact your MicroStrategy sales representative to determine the location and login credentials for the MicroStrategy download site.

- You may have to extract the downloaded files to locate the `Setup.exe` file. When extracting the files, ensure that the extraction software maintains the folder structure of the compressed files. Most extraction software maintains the folder structure by default, but if you use WinRAR, ensure that you select the Extract full paths option.

- To review an alternative, guided introduction to installing MicroStrategy software, you can locate and run the `MICROSTRATEGY.exe` file. For information on this installation alternative, see *Installing with a guided MicroStrategy introduction, page 109*.

If this is the first time you have installed MicroStrategy, you are prompted to choose the language for the wizard. Select the appropriate language from the drop-down list and click **OK**.

The MicroStrategy Installation Wizard opens and leads you through the rest of the installation process. The sections below describe the actions you must take for each page in the wizard. After you enter all required information on an installation page, click **Next** to proceed to the next page of the installation routine.

If any services are running for previously installed MicroStrategy products, you are prompted to stop them. Click **Yes** to proceed. If you click **No**, you cannot install MicroStrategy products until you stop all MicroStrategy services.
**Welcome**

Read the information on the welcome screen and proceed to the next step.

If you opened the MicroStrategy Installation Wizard through the Microsoft Control Panel using the Add/Remove Programs option, the wizard opens the Welcome page in maintenance mode. For more information on modifying, repairing, or removing all or part of your MicroStrategy installation, see *Chapter 13, Adding or Removing MicroStrategy Components*. 

**License Agreement**

Read the license agreement, and accept or decline the agreement by clicking the appropriate button. If you decline, you cannot install MicroStrategy products.

Click **Print** to print a copy of the license agreement for your records.

**Customer Information**

Enter the following customer information:

- **User Name**
- **Company Name**
- **License Key**

Licensed users can contact Technical Support to obtain a license key.

**Choose Destination Location**

Browse to the locations where the MicroStrategy products and MicroStrategy common files are to be installed:

- **MicroStrategy Destination Folder**: Browse to and select the location where MicroStrategy products are installed. This is where executable files and other support files are installed for your licensed MicroStrategy products.
While this setting determines the default root directory for the MicroStrategy products you install, you can change the destination of an individual product later as part of selecting which MicroStrategy products to install.

You can choose the directory for a product only if that product is not already installed on the server machine. Otherwise, the product can only be installed in the same directory in which it already exists.

- **MicroStrategy Common Files Destination Folder**: Browse to and select the location where MicroStrategy common files are installed. These files are required to support a MicroStrategy installation.

### Select Features

You can select the MicroStrategy products to install as well as define the location in which to install the products.

Once you have selected all required MicroStrategy products and defined the proper installation locations, if you are prompted to stop your MicroStrategy Web server, click **Yes**. If you click **No**, you cannot continue with the installation until you stop your MicroStrategy Web server.

### Installing MicroStrategy products

Select the check box next to a MicroStrategy product to include that product in the installation. Alternatively, you can clear a check box to uninstall or exclude a MicroStrategy product from the installation.

The installation pages you see after this step depend on the products you choose to install. These instructions describe all possible pages to support all products of the MicroStrategy Product Suite. You do not have to install all of these products on the same machine. In fact, this is strongly discouraged in a production environment. For basic guidelines about product deployments, see *Recommended installation location and example deployments, page 20* in *Chapter 1, Planning Your Installation*.

Depending on your license key, you can install the Universal (platform independent) version of some of the products listed below. For example, instead of MicroStrategy Web you may see MicroStrategy Web Universal.
Installation and Configuration Guide

Installing MicroStrategy on Windows

© 2011 MicroStrategy, Inc.
Installation procedure

- MicroStrategy Intelligence Server (see MicroStrategy Intelligence Server, page 5)

- MicroStrategy Web (see MicroStrategy Web and Web Universal, page 8)
  - MicroStrategy Portlets (see MicroStrategy Portlets, page 10)
  - MicroStrategy GIS Connectors (see MicroStrategy GIS Connectors, page 11)

- MicroStrategy Office (see MicroStrategy Office, page 15)
  - If you are a MicroStrategy Web administrator, you can allow Web users to install MicroStrategy Office by making an ‘Install MicroStrategy Office’ link available in MicroStrategy Web. When a user chooses to install MicroStrategy Office, MicroStrategy Office is installed as a stand-alone product on his or her machine. MicroStrategy Office can be installed even if no other MicroStrategy products are available on his or her machine. For steps to enable users to install MicroStrategy Office from Web, see Enabling users to install MicroStrategy Office from Web, page 287 of Chapter 6, Deploying MicroStrategy Web and Web Universal.

- MicroStrategy Mobile (see MicroStrategy Mobile, page 12)

- MicroStrategy Desktop Products (see MicroStrategy Desktop products, page 3)

- MicroStrategy Object Manager (see MicroStrategy Object Manager, page 4)

- MicroStrategy Command Manager (see MicroStrategy Command Manager, page 4)

- MicroStrategy Enterprise Manager (see MicroStrategy Enterprise Manager, page 5)

- MicroStrategy Integrity Manager (see MicroStrategy Integrity Manager, page 15)

- MicroStrategy SDK (see MicroStrategy SDK, page 13)
• MicroStrategy Narrowcast Server (see MicroStrategy Narrowcast Server, page 17)

• MicroStrategy Analytics Modules (see MicroStrategy Analytics Modules, page 18)

• MicroStrategy Tutorial - Reporting (see MicroStrategy Tutorial Reporting, page 18)

• Other components:

  □ Sequelink ODBC Socket Server is required to support MicroStrategy Narrowcast Server. It can also be used to access Microsoft Access databases and Microsoft Excel files stored on a Windows machine from an Intelligence Server hosted on a UNIX or Linux machine (see MicroStrategy ODBC Driver for SequeLink, page 475).

  □ TM1 Connector for MicroStrategy is required to connect to IBM Cognos TM1 data sources. For information on connecting to TM1 data sources, see the MDX Cube Reporting Guide.

Many of the platform components have subcomponents. If you expand the different MicroStrategy products, you can select the appropriate check boxes to specify the subcomponents to install. For information on MicroStrategy components and subcomponents, see MicroStrategy components, page 2 in Chapter 1, Planning Your Installation.

**Defining the installation location for MicroStrategy products and sub-components**

You can select MicroStrategy products and their sub-components to define their installation locations. When you select a MicroStrategy product or sub-component, the Destination Folder area near the bottom of the interface displays the current installation folder for the product. Click Browse to select a different installation folder.

If you select a MicroStrategy product or sub-component and the Browse button is not accessible, this means that the installation location cannot be changed. For example, if you select MicroStrategy Office you cannot define a installation location. However, if you expand this product, you can define the installation location for its subcomponents.
**Server Activation**

If you install one or more MicroStrategy Server products, you can request an Activation Code to activate your MicroStrategy Server products upon completion of the installation process. MicroStrategy Server products include:

- MicroStrategy Intelligence Server
- MicroStrategy Web or Web Universal
- MicroStrategy Narrowcast Server
- MicroStrategy Integrity Manager

The next few pages of the installation process guide you in providing the information you must submit to MicroStrategy to request an Activation Code.

**Welcome**

Read the information on the Welcome screen and click **Next** to proceed to the next step.

**Server Information**

Specify information about your Intelligence Server installation. Enter the following characteristics:

- **Name**: distinguishes the name of this Intelligence Server installation from any other Intelligence Server installations in your company
- **Location**: physical location of the machine on which Intelligence Server is installed
- **Use**: description of how Intelligence Server is used

Click **Privacy Statement** to view the MicroStrategy Privacy Statement.

**Installer Information**

Specify contact information of the person installing the software. After your installation is complete an email containing the Activation Code is sent to the
email address you confirm in this software activation step. Enter the following installer information:

- Specify whether you are an employee of the licensed company or installing on behalf of the licensed company.

- For descriptions of what information to include in the other text fields, press F1 to view the MicroStrategy online help.

Note the following:

- Select the check box at the bottom of the page to receive notifications about product updates, events, and special offers from MicroStrategy.

- Click Privacy Statement to view the MicroStrategy Privacy Statement.

Contact Information

You see this page if you indicated that you are not an employee of the company licensed to use this software, and are installing the software on behalf of that company.

Specify contact information for the employee licensed to use the software. After your installation is complete an email containing the Activation Code is sent to the email address you confirm in this software activation step. For descriptions of what information to include in the text fields, press F1 to view the MicroStrategy online help.

Note the following:

- Select the check box at the bottom of the page if you want to receive notifications about product updates, events, and special offers from MicroStrategy.

- Click Privacy Statement to view the MicroStrategy Privacy Statement.

Request Activation Code

This page includes options to request an Activation Code now or at a later time. This page provides the following options:

- Select Yes, I want to request an Activation Code and click Next to request an Activation Code. The Activation Code is sent to the email
addresses specified in the Installer Information and Contact Information pages. This email is sent upon completion of the installation process.

- Select **No, I will request the Activation Code at a later time** and click **Next** to request an Activation Code at a later time.

If you choose to request an Activation Code at a later time, a message is displayed that instructs you how to request an Activation Code after the installation procedure is completed. For more instructions on requesting an Activation Code at a later time, see *Request an Activation Code*, page 132 in *Chapter 4, Activating Your Installation*.

You have a grace period of 30 calendar days to activate your installation. If you do not complete the activation before the grace period expires, your MicroStrategy products stop functioning until you activate it. If you wait to activate your installation, you receive periodic reminders.

Once your installation is complete and you request an Activation Code, an email is sent to the email addresses you specified in the Installer Information and Contact Information pages of the software activation procedure. The email provides instructions on how to use the requested Activation Code to activate your software. To activate your installation, you can also use the steps given in *Activate your installation, page 134 in Chapter 4, Activating Your Installation*.

When the Activation Code request process is finished, you are prompted to either view the readme file or go directly to the MicroStrategy Installation Wizard Complete page. Click **Yes** to read the readme file or **No** to go to the MicroStrategy Installation Wizard Complete page.

**CPU License Information**

You see this page only if both of the following statements are true:

- You are installing MicroStrategy Intelligence Server on a multi-processor machine.
- Your license is based on CPU and allows for more than one CPU.

Specify the number of CPUs that Intelligence Server is licensed to use.
**MicroStrategy Web (ASP.NET) Setting**

You see this page only if you choose to install MicroStrategy Web (ASP.NET) and only if you do not have a previous version of MicroStrategy Web installed.

Specify the Internet Information Services (IIS) virtual directory to be created for MicroStrategy Web pages. The default is MicroStrategy. In IIS, a virtual directory is the home location for a set of Web pages that the Web server hosts.

**Note the following:**

- If you have a previous version of MicroStrategy Web installed on the machine, the new version you install uses the same virtual directory the previous version is using. Therefore, you are not prompted to specify the name of the virtual directory.

- The name provided for a virtual directory must be unique. You cannot use the same name as the default for other MicroStrategy products.

- MicroStrategy automatically installs the version of .NET Framework required for MicroStrategy Web, if that version of .NET Framework cannot be located on your installation machine. MicroStrategy also automatically configures the MicroStrategy Web virtual directory to run with the version of .NET Framework that it requires.

**MicroStrategy Web (ASP.NET) CPU Affinity Setting**

You see this page only if you choose to install MicroStrategy Web (ASP.NET) and if the MicroStrategy Web installation detects that the license key entered is a CPU-based license. This page is not displayed on single-processor machines.

Specify the number of CPUs that MicroStrategy Web is licensed to use on the machine. You can specify only the number of CPUs that are allowed by the license. If MicroStrategy Web is installed on more than one machine, the total number of CPUs should not exceed the maximum number of CPUs specified by the license. For machines that support hyper threading technology, the CPU counts correspond to physical CPUs, not logical CPUs.

**To allow the setting to take effect, the installation stops IIS. After IIS has been restarted, the MicroStrategy Web application uses the specified number of CPUs.**
For more information on the MicroStrategy Web CPU affinity feature, refer to the MicroStrategy System Administration Guide.

**MicroStrategy Mobile Server (ASP.NET) Setting**

You see this page only if you choose to install MicroStrategy Mobile Server (ASP.NET) and only if you do not have a previous version of MicroStrategy Mobile Server installed.

Specify the Internet Information Services (IIS) virtual directory to be created for MicroStrategy Mobile Server. The default is MicroStrategyMobile. The virtual directory is part of the URL used to access the interactive reporting and analysis applications deployed on this machine via Mobile Server.

Mobile Server can be deployed using the same techniques used to deploy MicroStrategy Web and Web Universal, as described in Deploying MicroStrategy Web and Web Universal, page 209. For additional configurations required to deploy Mobile Server, see the MicroStrategy Mobile Design and Administration Guide.

Note the following:

- If you have a previous version of MicroStrategy Mobile Server installed on the machine, the new version you install uses the same virtual directory the previous version is using. Therefore, you are not prompted to specify the name of the virtual directory.

- The name provided for a virtual directory must be unique. You cannot use the same name as the default for other MicroStrategy products.

- MicroStrategy automatically installs the version of .NET Framework required for MicroStrategy Mobile Server, if that version of .NET Framework cannot be located on your installation machine. MicroStrategy also automatically configures the MicroStrategy Mobile Server virtual directory to run with the version of .NET Framework that it requires.

**MicroStrategy Subscription Portal Setting**

You see this page only if you choose to install MicroStrategy Subscription Portal, which is a component of Narrowcast Server, and only if you do not have a previous version of Subscription Portal installed.
Specify the name of the IIS virtual directory to be created for MicroStrategy Subscription Portal pages. The default is NarrowcastServer. In IIS, a virtual directory is the home location for a set of Web pages that the Web server hosts.

Subscription Portal offers you the ability to subscribe to and view Narrowcast Server services, service descriptions, and their most recent modification dates on the Web. For complete information about Subscription Portal and other components of Narrowcast Server, refer to the MicroStrategy Narrowcast Server documentation.

The name provided for a virtual directory must be unique. You cannot use the same name as the default for other MicroStrategy products.

**MicroStrategy Web Services Setting**

You see this page only if you choose to install MicroStrategy Web Services, which is required to run MicroStrategy Office, and only if you do not have a previous version of Web Services installed.

Specify the IIS virtual directory to be created for MicroStrategy Web Services pages. The default is MicroStrategyWS. In IIS, a virtual directory is the home location for a set of Web pages that the Web server hosts.

The name provided for a virtual directory must be unique. You cannot use the same name as the default for other MicroStrategy products.

For information about deploying MicroStrategy Web Services ASP.NET and J2EE, see Chapter 7, *Deploying MicroStrategy Web Services*.

To learn more about MicroStrategy Web Services, refer to the *MicroStrategy Web Services Administration Guide*.

**MicroStrategy Web MMT Setting**

You see this page only if you choose to install MicroStrategy Web MMT and if you do not have a previous version installed.

In previous releases, MicroStrategy Web MMT was referred to as MicroStrategy eTrainer.
Specify the IIS virtual directory to be created for MicroStrategy Web MMT pages. The default is `WebMMT`. In IIS, a virtual directory is the home location for a set of Web pages that the Web server hosts.

The name provided for a virtual directory must be unique. You cannot use the same name as the default for other MicroStrategy products.

**TM1 Connector for MicroStrategy Setting**

You see this page only if you choose to install the TM1 Connector for MicroStrategy and if you do not have a previous version installed.

Specify the virtual directory to be created the TM1 Connector for MicroStrategy. The default is `MicroStrategyXmla`. This virtual directory is used as part of the URL to connect to TM1 data sources for integration with MicroStrategy. For information on connecting to TM1 data sources, see the *MDX Cube Reporting Guide*.

**Select Program Folder**

Specify the folder in your Windows Start menu from which MicroStrategy products are to be accessed. It is recommended that you accept the default program folder.

**MicroStrategy Intelligence Server Setting**

You see this page if you choose to install MicroStrategy Intelligence Server, and if you do not have a previous version of Intelligence Server installed.

Select the check box to use the local system account as the Intelligence Server service account. If you clear the check box to set a different Intelligence Server service account, enter the following information:

- **Login**: a Windows login of the form `Domain\User` with full administrative privileges under which to run the Intelligence Server service

  The user account used to run Intelligence Server must have full administrator privileges for the local machine. If the administrator default privileges have been modified for the user account, connection errors can occur. For example, if the user account is denied access to the DSN accessed by Intelligence Server, Intelligence Server connection fails.
• **Password**: a valid password for the Windows login entered in the Login box

• **Confirmation**: retype the password to confirm it is correct

If the password you supply changes, you must reconfigure the Windows service to use the new password. Otherwise, Intelligence Server connections fail when the connection attempts to authenticate the login and password.

**MicroStrategy Narrowcast Server Setting**

You see this page if you choose to install MicroStrategy Narrowcast Server, and if you do not have a previous version of Narrowcast Server installed.

Select the check box to bypass the creation of a Narrowcast Server service account.

It is recommend you create the Narrowcast Server service account. Clear the check box, and enter the following information:

• **Login**: a Windows login of the form `Domain\User` with administrative privileges under which to run the Narrowcast Server service

• **Password**: a valid password for the Windows login entered in the Login box

• **Confirmation**: retype the password to confirm that it is correct

If you change the password for this account, you must reconfigure the Narrowcast Server Windows services to use the new password.

Refer to the *MicroStrategy Narrowcast Server Installation and Configuration Guide* for additional details about this setting.

**MicroStrategy Office URL Setting**

You see this page if you choose to install MicroStrategy Office, and if you do not have a previous version of MicroStrategy Office installed.

Specify the URL for MicroStrategy Web Services. The MicroStrategy Office client requires the MicroStrategy Web Services URL to access MicroStrategy projects. The URL depends on the name of the IIS virtual directory that you
specified on the MicroStrategy Web Services page. To review the step in which the MicroStrategy Web Services page was specified, see MicroStrategy Web Services Setting, page 104.

Assuming that you kept the default value on the MicroStrategy Web Services page and you are installing on the same Web server machine that is hosting MicroStrategy Web, you should use the default URL provided:

http://localhost/MicroStrategyWS/MSTRWS.asmx

**MicroStrategy Office Configuration**

You see this page if you choose to install MicroStrategy Office and if you do not have a previous version of MicroStrategy Office installed.

Select the check boxes to enable MicroStrategy Office for the associated Microsoft applications. You can configure MicroStrategy Office to integrate with Microsoft Excel, PowerPoint, or Word. The MicroStrategy Office toolbar is added to the Microsoft Office applications that you select.

**Start Copying Files**

This page displays the following information about your installation:

- Products that will be installed or updated
- Target directories in which the products are installed
- Name of the Windows Start menu program folder
- Virtual directories for MicroStrategy Web (ASP.NET), Narrowcast Server Subscription Portal, Web Services, and Web MMT
- URL for MicroStrategy Web Services
- Service accounts for MicroStrategy Narrowcast Server and Intelligence Server
- Location of the installation log file
- License details

Click **Install** to continue with the installation process, which can take several minutes depending on your computer’s hardware configuration.

Click **Print** to print a copy of this information for your records.
MicroStrategy Installation Wizard Complete

If the option to restart your machine appears, select **Yes I want to restart my computer now** to ensure that the installation process finishes correctly. This is the recommended procedure, but you can also choose to continue without restarting.

Click **Finish** to complete the installation.

If you encounter errors while installing MicroStrategy, refer to *Appendix B, Troubleshooting.*

Installation verification

During the installation routine, the MicroStrategy Installation Wizard gathers and records information about your system and your installation selections. You can verify installation setup information through the installation log file (*install.log*), located by default in:

- **32-bit Windows environments:** `C:\Program Files\Common Files\MicroStrategy`.
- **64-bit Windows environments:** `C:\Program Files (x86)\Common Files\MicroStrategy`.

The installation log file includes the following information:

- Installation date
- Target directories
- Program folder name
- Operating system identification
- Hardware specifications
- Selected installation options
- Registry paths
- List of registered files

*The installation log file can be helpful if you encounter errors during the installation process. For example, the log can tell you if a registry key or path was not added or if a critical file was not registered successfully.*
Installing with a guided MicroStrategy introduction

The installation procedure provided in this chapter assumes that you use the Setup.exe file to install MicroStrategy software. As an alternative, a guided introduction to MicroStrategy software and the installation process is also provided. This introduction is provided as an Adobe Flash visualization.

To use this Flash visualization, you must locate and run the file MICROSTRATEGY.exe, which is available in the MicroStrategy install media or the files downloaded from the MicroStrategy download site. You can then use the Flash visualization to review documentation on MicroStrategy software, as well as begin the installation process.

If you provide the MicroStrategy installation files on a network location, you must map a network drive for users to access the MICROSTRATEGY.exe file. If users run this file without locating it through the use of a mapped network drive, the links to open various product manuals will not function properly.

To continue with the installation procedure, see Welcome, page 95.

Configuring your MicroStrategy installation

After completing the MicroStrategy Installation Wizard steps to install MicroStrategy products, you can set up and configure your installation. To help guide the rest of your installation and configuration steps, refer to the section Installing and configuring MicroStrategy on Windows, page 87 in Chapter 1, Planning Your Installation, for an installation and configuration checklist.

Note the following:

• The next chapter in the installation and configuration checklist and in this guide covers software activation steps with MicroStrategy. These steps should be done before or soon after the Configuration Wizard tasks mentioned below. For more information, refer to Chapter 4, Activating Your Installation.

• After restarting your machine to complete an initial MicroStrategy installation, the MicroStrategy Configuration Wizard opens. The Configuration Wizard allows you to configure your MicroStrategy production environment. For more information, refer to Chapter 5, Configuring and Connecting Intelligence Server.
Introduction

This chapter describes the procedure for installing MicroStrategy on UNIX and Linux platforms. MicroStrategy products are compatible with Solaris, AIX, HP-UX, RedHat, and SUSE. The installation procedure described in this chapter refer to installing MicroStrategy on Solaris, AIX, HP-UX, RedHat, and SUSE. This chapter covers the following sections:

- *Installation procedures on UNIX and Linux*, page 112
- *Configuring your MicroStrategy installation*, page 129

Before installing MicroStrategy products, you should refer to *Chapter 1, Planning Your Installation* for important pre-installation information.

If you are installing MicroStrategy on Windows, refer to *Chapter 2, Installing MicroStrategy on Windows*.

Additionally, *Chapter 11, Deploying OEM Applications* explains the common workflow for deploying the MicroStrategy platform as an Original Equipment Manufacturer (OEM) application.
Installation procedures on UNIX and Linux

This section describes installing MicroStrategy on UNIX and Linux. The MicroStrategy products that you can install on UNIX and Linux environments are:

- MicroStrategy Intelligence Server Universal
- MicroStrategy Web Universal
- MicroStrategy Portlets
- MicroStrategy Web Services J2EE
- MicroStrategy Mobile
- MicroStrategy Command Manager
- MicroStrategy Integrity Manager
- MicroStrategy SDK

For more information about these products, see MicroStrategy components, page 2 in Chapter 1, Planning Your Installation.

It is recommended that you install MicroStrategy products as the root user.

Be aware of the following:

- If you are installing MicroStrategy products with a CPU-based license, you must be logged in as the root user; otherwise an error message is displayed and the installation fails.

- If you want a non-root user to be the administrator of the server, you must manually change the ownership after running the installation. Intelligence Server operation is dependent on root user privileges and permissions. Therefore, changing the ownership of Intelligence Server to a non-root user is not a certified or recommended practice.

- Only a user with root permissions can register MicroStrategy Intelligence Server as a service. If the Intelligence Server is registered as an application during installation, the root user can register the server as a service after running the installation. For more information on running Intelligence Server as a service, see the System Administration Guide.

- During installation, the user account for Intelligence Server is tested to verify that it can successfully support the use of common
system tools for the operating system. If you change the user account for Intelligence Server, you must verify that this user account can use and access common system tools for the operating system.

- **Script files** within `HOME_PATH/env` and other configuration files within `HOME_PATH` (see *Home Directory, page 117* for information on this MicroStrategy directory) are overwritten anytime a new MicroStrategy product is installed on a machine. Backup copies of the previous file are also created during the installation. These backup copies can be used to update the new versions of the script and configuration files to include any prior modifications.

For example, Intelligence Server is installed on a machine. Then a week later Command Manager is installed on the same machine. During this installation of Command Manager, script files such as `ODBC.sh` are overwritten and a backup copy of each of these files prior to the installation of Command Manager is also created.

### Different methods of installation

MicroStrategy products can be installed on UNIX and Linux, either in graphical user interface (GUI) mode or in command line mode, using the MicroStrategy Installation Wizard. In both cases, the MicroStrategy Installation Wizard runs, displaying the same pages and requesting the same information. The main differences are in how you provide the information and navigate through the wizard.

### Using GUI mode

The GUI (graphical user interface) mode presents a user interface for each page in the MicroStrategy Installation Wizard. The following navigational buttons are displayed:

- **Next**— proceed to the next page
- **Back**—return to the previous page
- **Cancel**—cancel the installation and close the MicroStrategy Installation Wizard
- **Finish**—complete the setup and close the wizard
Using command line mode

In command line mode, you type the appropriate information at the prompt and press ENTER. Instructions are included for each page of the MicroStrategy Installation Wizard.

In some cases, you are asked to make a selection by pressing 1 or 2, followed by pressing ENTER. You then press 0 and ENTER to continue.

Defaults appear next to each prompt and are enclosed in square brackets, for example, [1]. Press ENTER to use the default, or type a different response to the prompt to override the default.

In addition, on the command line wizard pages, the following options are available:

- Press 1 and then press ENTER to proceed to the next page.
- Press 2 and then press ENTER to return to the previous page.
- Press 3 and then press ENTER to cancel the installation and close the MicroStrategy Installation Wizard.
- On the last page, which is MicroStrategy Installation Wizard Complete, press 3 and then press ENTER to complete the setup and close the wizard.

Installing with the MicroStrategy Installation Wizard

To install MicroStrategy products, you must log on to your machine using a valid UNIX or Linux account. For ease of management and maintenance, it is recommended that you create a dedicated user account.

Note the following:

- You need root access permissions for installation if you have purchased the CPU-based MicroStrategy license.
- If you want to enable additional error and troubleshooting issue logging for the MicroStrategy installation routine, contact MicroStrategy Technical Support.

To exit the installation process at any time, click Cancel.
To access the MicroStrategy Installation Wizard

1 Log on to the machine on which you are installing one or more MicroStrategy products.

2 Browse to the MicroStrategy Installation folder. Depending on your UNIX or Linux environment, browse to one of the following folders:
   - **Solaris**: QueryReportingAnalysis_SunOS
   - **AIX**: QueryReportingAnalysis_AIX
   - **HP-UX**: QueryReportingAnalysis_HP-UX
   - **Linux**: QueryReportingAnalysis_Linux

   Note the following:
   - You can access the installation files from a disk or ask your system administrator to share the files on a network location. There are different disks for installing MicroStrategy products on different platforms: Windows, Solaris, AIX, HP-UX, and Linux. For information on mounting or unmounting CD-ROM drives, refer to *Mounting and unmounting CD-ROMs, page 526* in Appendix B, Troubleshooting.
   - If your CD-ROM drive is configured to display directories with short names in lowercase font the directory names are in the form instal 1/queryr 1/. You can still run the installation but it is not as easy to locate the correct directories. For information on mounting CD-ROM drives, refer to *Mounting and unmounting CD-ROMs, page 526* in Appendix B, Troubleshooting.

3 Type one of the following, depending on the installation mode you chose and your operating system:
   - To run the wizard in GUI mode
     - **Solaris**: ./setupsol.bin
     - **AIX**: ./setupAIX.bin
     - **HP-UX**: ./setupHPIA64.bin
     - **Linux**: ./setupLinux.bin
• To run the wizard in command line mode
  – **Solaris**: ./setupsol.bin -console
  – **AIX**: ./setupAIX.bin -console
  – **HP-UX**: ./setupHPIA64.bin -console
  – **Linux**: ./setupLinux.bin -console

  For more information on these modes of installation, refer to *Different methods of installation, page 113.*

4 The MicroStrategy Installation Wizard opens and leads you through the rest of the installation process. The following sections describe the actions you need to take for each page in the wizard.

![Note the following:](image)

- To complete the installation, you must have write permissions in the installation directory; otherwise the installation fails.
- After you enter all required information on an installation page, click **Next**, or press 1 and then press **ENTER**, to proceed to the next page of the installation routine.
- At any time during the setup, you can click **Cancel**, or press 3 and then press **ENTER**, to quit the installation.

**Language Setup**

Specify the language to be used for the MicroStrategy installation and proceed to the next step.

**Welcome**

Read the information on the welcome screen and proceed to the next step.

**License Agreement**

Read the license agreement, and select to accept or decline the agreement. If you choose to decline, you cannot install MicroStrategy products.
Customer Information

Enter the following customer information:

- **User**
- **Company**
- **License Key**

Licensed users can contact Technical Support to obtain a license key.

Home Directory

Specify the location where the MicroStrategy configuration files and application launchers are to be installed, according to the following guidelines:

- **The default location is** `/var/opt/MicroStrategy`, or `$HOME/MicroStrategy` if you do not have write access to `/var/opt/MicroStrategy`.

- **Do not install the MicroStrategy configuration files directly to your UNIX or Linux home directory ($HOME).** To ensure that the required permissions can be defined for the MicroStrategy configuration files, you must install these files within a separate directory. For example, the default path of `$HOME/MicroStrategy` uses the MicroStrategy directory within `$HOME` to ensure permissions on these files are defined correctly.

- **The path specified for the home directory is referred to as** `HOME_PATH` **in this guide.**

- **Do not change the names of folders within the** `HOME_PATH` **after installing Intelligence Server.**

- **When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.**
Install Directory

Specify the location where the MicroStrategy products are to be installed, according to the following guidelines:

- The default location is /opt/MicroStrategy, or $HOME/MicroStrategy/install if you do not have write access to /opt/MicroStrategy.

- The path specified for the install directory is referred to as INSTALL_PATH in this guide.

- Do not change the names of folders within the INSTALL_PATH after installing Intelligence Server.

- When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.

Log Directory

Specify the location where the MicroStrategy application logs are to be created, according to the following guidelines:

- The default location is /var/log/MicroStrategy, or $HOME/MicroStrategy/log if you do not have write access to /var/log/MicroStrategy.

- The path specified for the log directory is referred to as LOG_PATH in this guide.

- When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.

Select Components

Select the check box of a MicroStrategy product to include that product in the installation. Alternatively, you can clear a check box to uninstall or exclude a MicroStrategy product from the installation.

The installation pages you see after this step depend on the products you choose to install. These instructions describe all possible pages.

Many of the platform components have subcomponents. If you expand the different MicroStrategy products, you can select the appropriate check boxes to specify the subcomponents to install. For information on MicroStrategy
components and subcomponents, see *MicroStrategy components, page 2* in *Chapter 1, Planning Your Installation*.

You can only see MicroStrategy products that are available with your license key.

**System Requirements**

This page is displayed only if the machine you are installing Intelligence Server Universal on does not use the recommended system resource limits to support the use of shared memory resources. It is recommended that you exit the installation and configure these system settings to support shared memory resources. For information on this requirement and the options available to complete the installation, see *Configuring shared memory resources, page 53*.

**CPU License Information**

This page is displayed only if the Intelligence Server Universal license has a CPU number limitation.

Specify the number of CPUs that Intelligence Server Universal is licensed to use.

**Missing Requirements**

This page is displayed only if there are system requirements that are not met to install the MicroStrategy products you selected. Review the list of requirements to determine if you can proceed with the installation, or if the installation must be cancelled.

If you are installing MicroStrategy Intelligence Server Universal on Linux, you may see a warning about the value for the Linux kernel setting `vm.max_map_count`. For information on this setting and MicroStrategy’s recommendation on its value, see *Supporting Intelligence Server Universal memory allocation on Linux, page 58*.

To improve the performance of MicroStrategy Intelligence Server Universal for large scale production applications, Intelligence Server Universal can be configured to use shared memory resources. If a semaphore configuration warning is displayed, some system resource limits are not configured to fully
support the use of shared memory resources. To support this configuration, cancel the installation and refer to the limit recommendations provided in Configuring shared memory resources, page 53.

**Command Manager Install Location**

You see this page only if you choose to install MicroStrategy Command Manager.

Specify the location where MicroStrategy Command Manager files are to be installed. The default directory is INSTALL_PATH/CommandManager. When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.

**MicroStrategy Web Universal and MicroStrategy Web Server for Mobile JSP Install Location**

The name of this page changes to reflect which of these two components you have chosen to install. You see this page only if you choose to install MicroStrategy Web Universal and/or MicroStrategy Web Server for Mobile JSP.

Specify the directory where the MicroStrategy Web Universal web application archive (.war) file is to be installed. The default directory is INSTALL_PATH/WebUniversal. When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.

**BlackBerry Device Client Application Install Directory**

You see this page only if you choose to install MicroStrategy Mobile BlackBerry Device Client Application.

Specify the location where MicroStrategy Mobile BlackBerry Device Client Application files are to be installed. The default directory is INSTALL_PATH/Mobile/BlackberryClient. When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.
MicroStrategy Mobile Server JSP Install Location

You see this page only if you choose to install MicroStrategy Mobile Server JSP.

Specify the directory where the MicroStrategy Mobile Server JSP web application archive (.war) file is to be installed. The default directory is INSTALL_PATH/Mobile/MobileServer. When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.

Mobile Server can be deployed using the same techniques used to deploy MicroStrategy Web and Web Universal, as described in Chapter 6, Deploying MicroStrategy Web and Web Universal. For additional configurations required to deploy Mobile Server, see the MicroStrategy Mobile Design and Administration Guide.

MicroStrategy Portlets Install Location

You see this page only if you choose to install MicroStrategy Portlets.

Specify the location where MicroStrategy Portlet files are to be installed. The default directory is INSTALL_PATH/Portlets. When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.

MicroStrategy Web Services J2EE for MicroStrategy Office Install Location

You see this page only if you choose to install MicroStrategy Web Services J2EE.

Specify the location where MicroStrategy Web Services J2EE Web application archives (.war) files are to be installed. The default directory is INSTALL_PATH/WebServicesJ2EE. When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.

For information about deploying MicroStrategy Web Services ASP.NET and J2EE, see Chapter 7, Deploying MicroStrategy Web Services.
MicroStrategy SDK Installation Location

You see this page only if you choose to install MicroStrategy SDK.

Specify the directory where MicroStrategy SDK files are to be installed. The default directory is INSTALL_PATH/SDK. When including paths during a MicroStrategy installation, include absolute paths rather than relative paths.

Software Activation

If you have installed one or more MicroStrategy server products, you can request an Activation Code to activate your MicroStrategy server products upon completion of the installation process. MicroStrategy server products include:

- MicroStrategy Intelligence Server
- MicroStrategy Web or Web Universal
- MicroStrategy Narrowcast Server
- MicroStrategy Integrity Manager

The next few pages of the installation process guide you in providing the information you must submit to MicroStrategy to request an Activation Code.

Welcome

Read the information on the welcome screen and proceed to the next step.

Server Information

Specify information about your MicroStrategy server installation. Enter the following characteristics:

- **Name**: Distinguishes the name of this MicroStrategy server product installation from any other MicroStrategy server product installations in your company.

- **Location**: Physical location of the machine on which MicroStrategy server products are installed.
• **Use**: Description of how the server is used.

Click **Privacy Statement** to view the MicroStrategy Privacy Statement.

**Installer Information**

Specify contact information of the person installing the software. After your installation is complete an email containing the Activation Code is sent to the email address you confirm in this software activation step. Enter the following installer information:

- Specify whether you are an employee of the licensed company or installing on behalf of the licensed company.
- Enter the necessary data into all text fields. Make sure the email address you enter is correct. This email address is the recipient of the Activation Code.

Note the following:

- Select the check box at the bottom of the page to receive notifications about product updates, events, and special offers from MicroStrategy.
- Click **Privacy Statement** to view the MicroStrategy Privacy Statement.

**Contact Information**

You see this page if you indicated that you are not an employee of the company licensed to use this software, and are installing the software on behalf of that company.

Specify contact information for the employee license to use the software. Enter the necessary data into all text fields. Make sure the email address you enter is correct. After your installation is complete an email containing the Activation Code is sent to the email address you confirm in this software activation step.

Note the following:

- Select the check box at the bottom of the page to receive notifications about product updates, events, and special offers from MicroStrategy.
• Click Privacy Statement to view the MicroStrategy Privacy Statement.

Request Activation Code

This page includes options to request an Activation Code now or at a later time. This page provides the following options:

• Select Yes, I want to request an Activation Code and click Next to request an Activation Code. The Activation Code is sent to the email addresses supplied in the Installer Information and Contact Information pages.

• Select No, I will request the Activation Code at a later time and click Next to request an Activation Code at a later time.

If you choose to request an Activation Code at a later time, a message is displayed that instructs you how to request an Activation Code after the installation procedure is completed. For more instructions on requesting an Activation Code at a later time, see Request an Activation Code, page 132 in Chapter 4, Activating Your Installation.

You have a grace period of 30 calendar days to activate your installation. If you do not complete the activation before the grace period expires, your MicroStrategy product stops functioning until you activate it. If you wait to activate your installation, you receive periodic reminders.

Once you request an Activation Code, an email is sent to the email addresses you specify in the Installer Information and Contact Information pages of the software activation procedure. The email provides instructions on how to use the requested Activation Code to activate your software. To activate your installation, you can also use the steps given in Activate your installation, page 134 in Chapter 4, Activating Your Installation.

Start copying files

The Start Copying Files page displays the following information about your installation:

• Products that will be installed or updated
• Target directories in which the products are installed
• Location of the installation log file
• License details
When you proceed from this step, the installation process begins, which can take several minutes depending on your computer’s hardware configuration.

**File Transfer Complete**

Review the information about your installation and proceed to the next step.

**MicroStrategy Install Wizard Complete**

You can select to view the MicroStrategy readme and run MicroStrategy Configuration Wizard upon completion of your MicroStrategy installation. The Configuration Wizard allows you to configure your MicroStrategy production environment. For more information, refer to *Chapter 5, Configuring and Connecting Intelligence Server*.

Click **Finish**, or press 3 and then **ENTER**, to complete the installation.

**Unique post-installation configurations.**

MicroStrategy supports a large number of different UNIX and Linux environments with various system configurations. There are a few cases in which you must perform some manual configurations to support the use of MicroStrategy on your system.

- *Migrating Intelligence Server from Windows to UNIX or Linux*, page 125
- *Create links for Intelligence Server startup in SUSE Linux*, page 126
- *Supporting fonts for documents, exported reports, and graphs*, page 126

**Migrating Intelligence Server from Windows to UNIX or Linux**

If you are installing MicroStrategy Intelligence Server Universal on UNIX or Linux and previously had Intelligence Server installed on a Windows platform, it is strongly recommended you modify certain system tuning settings. These memory and cache settings govern and can optimize the performance of Intelligence Server Universal and MicroStrategy projects in your 64-bit UNIX or Linux environment. For more information on these system tuning steps, see the After the Upgrade chapter of the *MicroStrategy Upgrade Guide*. 
Create links for Intelligence Server startup in SUSE Linux

If you are installing Intelligence Server on a SUSE Linux environment, you must manually create links for some system files. If you do not create these links, Intelligence Server cannot start correctly.

You need root permissions to access the files and create the necessary links described in this section.

To manually create links for Intelligence Server startup

1. In a console window, browse to the system folder `usr/lib64`.
2. In a console window, create the link of `libssl.so.4` to
   `libssl.so.0.9.7` with the command,
   
   ```
   ln libssl.so.0.9.7 libssl.so.4
   ```
3. In a console window, create the link of `libcrypto.so.4` to
   `libcrypto.so.0.9.7` with the command,
   
   ```
   ln libcrypto.so.0.9.7 libcrypto.so.4
   ```

Supporting fonts for documents, exported reports, and graphs

When Intelligence Server Universal is running on a UNIX or Linux platform, all fonts are converted to the Courier New font for:

- Reports exported to PDF format
- Report Services documents
- Graphs contained in HTML documents
- Graphs displayed in MicroStrategy Web

This occurs because the fonts required by the PDF component are missing from UNIX and Linux machines running Intelligence Server Universal.

MicroStrategy cannot package these fonts with Intelligence Server Universal due to licensing restrictions.
For steps to support fonts such as Microsoft True Type fonts for the MicroStrategy features listed above, see Setup for executing Report Services documents, page 342.

### Verifying installation

During the installation routine, the MicroStrategy Installation Wizard gathers and records information about your system and your installation selections. You can verify the setup information through the installation log file (`install.log`).

By default, the log file is located in `INSTALL_PATH` where, `INSTALL_PATH` is the directory you specified as the install directory in the MicroStrategy Installation Wizard.

The log file includes information about the following:

- Installation date
- Target directory
- Operating system identification
- Selected installation options
- Selected licensing details

This log file can be helpful if you encounter errors during the installation process. The log file records the reasons due to which the errors occurred.

### Directory structure

The following table describes the directories in which MicroStrategy files are installed.

For an installation on HP-UX, the folders `INSTALL_PATH/bin32` and `INSTALL_PATH/lib32` are not created because HP-UX uses
64-bit ODBC drivers, which allows all the binary files to be in 64-bit format.

<table>
<thead>
<tr>
<th>PATH/Directory</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOME_PATH</td>
<td>Configuration files that can be modified after installation.</td>
</tr>
<tr>
<td>HOME_PATH/env</td>
<td>Scripts to set up the proper environment for the MicroStrategy applications. If additional products are installed on the same machine at a later time, backups of the original scripts are saved here.</td>
</tr>
<tr>
<td>HOME_PATH/bin</td>
<td>Scripts to launch the MicroStrategy applications.</td>
</tr>
<tr>
<td>INSTALL_PATH</td>
<td>Files that are not supposed to change after the installation is complete.</td>
</tr>
<tr>
<td>INSTALL_PATH/CommandManager</td>
<td>MicroStrategy Command Manager files. This is the default directory for Command Manager but another location can be selected during installation.</td>
</tr>
<tr>
<td>INSTALL_PATH/GISConnectors</td>
<td>MicroStrategy Portlet files. This is the default directory for the Portlets but another location can be selected during installation.</td>
</tr>
<tr>
<td>INSTALL_PATH/IntelligenceServer/bin</td>
<td>Intelligence Server-specific binary files.</td>
</tr>
<tr>
<td>INSTALL_PATH/Mobile</td>
<td>MicroStrategy Mobile and Mobile Server JSP files. This is the default directory for Mobile but another location can be selected during installation.</td>
</tr>
<tr>
<td>INSTALL_PATH/PDFGeneratorFiles</td>
<td>Support files (fonts) for the PDF generation feature of Intelligence Server.</td>
</tr>
<tr>
<td>INSTALL_PATH/ReleaseNotes</td>
<td>Release notes and readme files for this release of MicroStrategy products.</td>
</tr>
<tr>
<td>INSTALL_PATH/Portlets</td>
<td>MicroStrategy Portlet files. This is the default directory for the Portlets but another location can be selected during installation.</td>
</tr>
<tr>
<td>INSTALL_PATH/SDK</td>
<td>Software Development Kit files. This is the default directory for the SDK but another location can be selected during installation.</td>
</tr>
<tr>
<td>INSTALL_PATH/WebUniversal</td>
<td>MicroStrategy Web Universal deployment path.</td>
</tr>
<tr>
<td>INSTALL_PATH/_jvm</td>
<td>The Java Runtime Environment (JRE) to be used by the Java applications. It provides the requirements for executing a Java application, a Java Virtual Machine, core classes, and supporting files.</td>
</tr>
<tr>
<td>INSTALL_PATH/_uninst</td>
<td>Launch files for uninstalling MicroStrategy Universal.</td>
</tr>
<tr>
<td>INSTALL_PATH/bin</td>
<td>64-bit binary files.</td>
</tr>
<tr>
<td>INSTALL_PATH/bin32</td>
<td>32-bit binary files.</td>
</tr>
</tbody>
</table>
Configuring your MicroStrategy installation

After completing the steps to install MicroStrategy products, you can set up and configure your installation. To help guide the rest of your installation and configuration steps, refer to the section *Installing and configuring MicroStrategy on UNIX and Linux*, page 89 in *Chapter 1, Planning Your Installation*, for an installation and configuration checklist.

<table>
<thead>
<tr>
<th>PATH/Directory</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>INSTALL_PATH/help</td>
<td>Online help files.</td>
</tr>
<tr>
<td>INSTALL_PATH/jar</td>
<td>Java libraries.</td>
</tr>
<tr>
<td>INSTALL_PATH/lib</td>
<td>64-bit binary libraries.</td>
</tr>
<tr>
<td>INSTALL_PATH/lib32</td>
<td>32-bit binary libraries.</td>
</tr>
<tr>
<td>INSTALL_PATH/locale</td>
<td>ODBC support messages. The ODBC drivers are copied to /lib32.</td>
</tr>
<tr>
<td>LOG_PATH</td>
<td>MicroStrategy application log files, which includes Intelligence Server log files.</td>
</tr>
</tbody>
</table>
INTRODUCTION

After your MicroStrategy installation is complete, you have 30 days to activate your installation. Before you activate your installation you must request an Activation Code from MicroStrategy. You can complete this request when you install MicroStrategy with the MicroStrategy Installation Wizard or after the installation using MicroStrategy License Manager.

This chapter describes the following procedures:

- Request an Activation Code
- Activate your installation

For answers to commonly asked questions about server activation, see Server Activation FAQ.
Request an Activation Code

You can request an Activation Code by supplying MicroStrategy with important information related to your installation. The information you provide helps MicroStrategy understand how you plan to use MicroStrategy software. With this information MicroStrategy can provide better information and technical support for your software configuration.

On Windows, MicroStrategy products can be activated only in graphics user interface (GUI) mode, using License Manager.

MicroStrategy products can be activated on UNIX and Linux, either in GUI mode or in command line mode, using License Manager. In both cases, License Manager runs and requests the same information. The main differences are in how you provide the information and navigate through the Activation Code request steps.

---

**Request an Activation Code with License Manager**

If you requested an Activation Code during installation you can skip this procedure and activate your installation by following the instructions in the next procedure, *Activate your software installation, page 134*.

1. Open MicroStrategy License Manager:

   - **Windows**: From the Start menu, point to Programs, then MicroStrategy, and then select License Manager. License Manager opens.

   - **UNIX/Linux**: License Manager can be run in GUI mode or command line mode.
     - **GUI**: In a UNIX or Linux console window, browse to HOME_PATH where HOME_PATH is the directory you specified as the home directory during installation. Browse to the folder bin and type ./mstrlicmgr, then press ENTER. The MicroStrategy License Manager opens in GUI mode.
     - **Command line**: In a UNIX or Linux console window, browse to HOME_PATH where HOME_PATH is the directory you specified as the home directory during installation. Browse to the folder bin
and type `./mstrlicmgr -console`, then press ENTER. The MicroStrategy License Manager opens in command line mode.

The steps to request an Activation Code in command line mode of License Manager vary from the steps below. Refer to the License Manager command line prompts to guide you through the steps to request an Activation Code. For more information specific to requesting an Activation Code for your installation in command line mode, refer to MicroStrategy Tech Note TN13550.

2 Select the License Administration tab. Under Server Activation select the Activate Server Installation option and click Next to continue to the next page.

3 Select the Generate Activation File and Request Activation Code option and click Next to continue to the next page.

4 Enter the characteristics of your server installation and click Next to continue to the next page.

5 Enter the contact information for the person who installed the software. Make sure to correctly select whether you are an employee of the licensed company or installing the software on the licensed company’s behalf.

   • If you select that you are an employee of the licensed company, click Next to continue to the next page. Once you complete the following step, the Activation Code is sent to the email address given; therefore it is important that the email address is valid and entered correctly.

   • If you select that you are not an employee of the licensed company, a contact information page is displayed after you click Next. Enter the contact information for the licensed company. Click Next to continue to the next page. Once you complete the following step, the Activation Code is sent to the email address given; therefore it is important that the email address is valid and entered correctly.

6 Select Yes, I want to request an Activation Code now and click Next.

An email containing the Activation Code is sent to the email address or addresses you confirmed in the steps above.
Activate your installation

After you have requested an Activation Code, MicroStrategy sends an email to the addresses provided in the request. This email contains the Activation Code that is necessary to complete the activation of your installation.

Activate your software installation

This procedure assumes that you have requested an Activation Code and received an email from MicroStrategy containing the Activation Code.

1. Open MicroStrategy License Manager:

   - **Windows**: From the Start menu, point to Programs, then MicroStrategy, and then select License Manager. License Manager opens.

   - **UNIX/Linux**: License Manager can be run in GUI mode or command line mode.
     - **GUI**: In a UNIX or Linux console window, browse to $HOME_PATH$ where $HOME_PATH$ is the directory you specified as the home directory during installation. Browse to the folder bin and type ./mstrlicmgr, then press ENTER. The MicroStrategy License Manager opens in GUI mode.
     - **Command line**: In a UNIX or Linux console window, browse to $HOME_PATH$ where $HOME_PATH$ is the directory you specified as the home directory during installation. Browse to the folder bin and type ./mstrlicmgr -console, then press ENTER. The MicroStrategy License Manager opens in command line mode.

   The steps to activate your installation in command line mode of License Manager vary from the steps below. Refer to the License Manager command line prompts to guide you through the steps to activate your installation. For more information specific to activating your installation in command line mode, refer to MicroStrategy Tech Note TN13550.

2. Select the License Administration tab. Under Server Activation select the Activate Server Installation option and click Next to continue to the next page.

   The step above is not necessary for License Manager in command line mode.
3 Select the **Server Activation using Activation Code** option and enter your Activation Code in the text field. Click **Next** to activate your software installation.

4 A verification message is displayed, click **OK** to close it.

You must restart your Intelligence Server for the activation status to update. You must also restart your Web server for the activation status to update in MicroStrategy Web.

### Configuring your MicroStrategy installation

After completing the steps to activate your installation, you can continue your setup and configuration. To help guide the rest of your installation and configuration steps, refer to the section **Installation and configuration checklists, page 87** in **Chapter 1, Planning Your Installation**, for installation and configuration checklists.

### Server Activation FAQ

#### What is MicroStrategy Server Activation?

MicroStrategy Server Activation is a licensing technology that ensures that installations of MicroStrategy server products are authentic and have been legitimately licensed. Server Activation registers each Server Installation with MicroStrategy and locks the installation to a specific machine.

#### Why is Server Activation necessary?

Server Activation provides benefits to both MicroStrategy and its customers:

- It ensures that the software products being used are authentic.
- It helps customers in identifying software installations to prevent over-installation.
- It improves customer service by maintaining a register of the hardware configurations used by our customers.
Does MicroStrategy adhere to Software Activation common practices?

Yes. Extensive research was performed on software activation and it was found that the Business Software Alliance (www.bsa.org) provides the most comprehensive information. MicroStrategy has closely followed these best practices.

Which products require activation?

All MicroStrategy modular and non-modular server products require server activation, including:

- Intelligence Server, Intelligence Server Module, Clustering Option, Report Services, Report Services Option, OLAP Services, OLAP Services Option, Distribution Services, MultiSource Option
- Mobile Server
- Narrowcast Server Delivery Engine

If more than one server product is installed on the same machine, does each server product need to be activated and deactivated separately?

No. All MicroStrategy server products installed on a single machine are grouped as a Server Installation and will be activated and deactivated as a group.

Which customers are required to activate MicroStrategy server products?

All customers who install MicroStrategy server products will need to activate their Server Installations.
Is a new CD Key needed to install products on different machines?

No, Server Activation has no impact on CD Keys. The latest CD Key sent by MicroStrategy can be used to install products on many machines just as before. The only difference now, is that each installation on a different machine will need a different Activation Code to activate the installation. Server Activation is independent of the CD Keys.

Is Server Activation required for both Named-user and CPU based licenses?

Yes. Server Activation is required for both Named-user and CPU based licenses. Server Activation is designed to track software installations regardless of licensing model. Server Activation does not manage or limit the number of servers on which server software is installed.

Does Server Activation aggregate the total number of licenses installed and prevent over-installation of products?

No. Server Activation and the licensing models are independent. However, using Server Activation information available at https://licensing.microstrategy.com, organizations can monitor the number of installations. Deactivating Server Installations that are not being used will ensure this information is up-to-date.

Does Server Activation apply to Evaluation Editions?

Yes. The Evaluation Edition must be activated within 7 days of installation.

What is the procedure for activating Server Installations?

Installing, modifying or upgrading MicroStrategy Server Installations will automatically generate an Activation XML file that contains information about the installation. This XML file, called the “Activation File” is uploaded to MicroStrategy either automatically by the installation routine or through License Manager; or by manually uploading the Activation File through a web browser via a secure web site, https://licensing.microstrategy.com. MicroStrategy then creates
a machine-specific Activation Code which is emailed to the installer and to the MicroStrategy Tech Support liaisons. The Activation Code must be manually entered into License Manager on the target Server Installation.

**Can the Server Installation be automatically activated after automatically requesting an activation code?**

No, the Activation Code is sent to the installer and to the MicroStrategy Tech Support contacts by email. Upon receiving the activation code, the Server Installation needs to be manually activated by entering the activation code using License Manager. For manual activations, the Activation Code can be copied from the Activation website, and pasted into License Manager.

**What information is sent to MicroStrategy in the Activation XML File?**

The following information is sent to the Activation XML File:

- **Installation Information:**
  - Installation timestamp
  - Activation ID (if the installation has previously been activated)
  - Contract information
  - CD Key used in installation
  - Installer contact details – name, address, email
  - Company contact details – name, address, email
  - Server Installation information – name, location, use
- **Hardware information:**
  - Unique Hardware Identifier
  - CPU Information – type, bit-size, clock speed, total quantity
  - Physical Memory installed
- **Operating System information:**
  - Type, version, bit-size, page/swap size
  - Locale
Additional information – 4GT mode and hyperthreading in Windows, LPAR ID, number of pooled CPUs, SMT used in AIX

- MicroStrategy information:
  - Install Type – new/modify/upgrade
  - Products and version installed
  - Number of CPUs allocated for CPU licenses

- Database information:
  - Metadata database and ODBC driver
  - Data warehouse database and ODBC driver

**Is the information sent to MicroStrategy secure?**

Yes. The Activation XML file is sent to MicroStrategy in the following ways:

- Automatically during installation or through License Manager. This information is encrypted using a RIPEMD-160 algorithm before being sent to MicroStrategy.
- Manually in the [https://licensing.microstrategy.com](https://licensing.microstrategy.com) website. Communication with this website is conducted through Secure Socket Layer once the user has been authenticated.

**Can I change the information in the Activation XML file?**

The content of this file is secured with a digital signature. Contact MicroStrategy Technical Support if the content is incorrect.

**What is used to lock the server to a machine?**

A unique hardware identifier for each machine is used to lock an installation to that machine. Any changes to these identifiers will require reactivation of the Server Installation:

- Windows: An identifier generated from a one-way hash of the network interface card MAC address
- AIX: Machine ID
• Solaris: Host ID

• RedHat Linux: An identifier generated from a one-way hash of the network interface card MAC address

• HP-UX: Exposes a unique Machine ID that can be obtained from the Operating System command line, using the following command:

  `getconf CS_MACHINE_IDENT`

Is there a grace period from the time server products are installed to when it can be activated?

Yes. There is a 30 calendar day grace period from installation (7 days for Evaluation Edition) before a server installation must be activated.

What happens if the Server Installations are not activated?

If a server installation has not been activated within 30 calendar days (7 days for Evaluation Edition), the server products will not be able to be restarted.

Should the installations be automatically or manually activated and deactivated?

It is preferable to automatically send the activation information to MicroStrategy, either during installation or using License Manager. This is an easy process that should take less than one minute to complete, and ensures that the correct information is sent to MicroStrategy.

What should be done if requesting an Activation Code fails during installation?

Firstly, allow the installation to complete. The Activation Code request should then be attempted using License Manager. If this does not work, activate the Server Installation manually by visiting the Activation website at [https://licensing.microstrategy.com](https://licensing.microstrategy.com).
What should be done if automatic request for an Activation Code does not work at all?

The Server Installation should be manually activated using License Manager on a machine that has access to the Web. The Activation XML file should be copied from the Server Installation that requires activation to this computer. If this does not work, contact MicroStrategy Technical Support.

Can the Activation Code be used on a different machine?

No. The Activation Code contains the unique identifier for a specific machine and can only be used on that machine. The server products will not be activated if the incorrect Activation Code is used.

What if the Server installation has to be moved to another machine?

If the Server Installation needs to be moved to another machine, uninstall the MicroStrategy products or deactivate the Server Installation from License Manager. Notify MicroStrategy that Server Installation has been deactivated. This keeps your active server inventory up to date as shown to you on https://licensing.microstrategy.com.

What if the server machine has a catastrophic failure and cannot be deactivated automatically or manually?

If a machine has a catastrophic failure and the server products cannot be uninstalled or deactivated, contact MicroStrategy Technical Support to update the status of this Server.

If there is more than one server product installed on a machine and one is removed, does the Server Installation need to be deactivated?

Upon removal of a server product, all remaining server products are automatically deactivated. The remaining server product(s) need to be reactivated to reflect the new product configuration on that machine. For example, if a machine contains MicroStrategy Narrowcast Server and
MicroStrategy Web, and MicroStrategy Narrowcast Server is removed, MicroStrategy Web will be automatically deactivated. MicroStrategy Web will need to be reactivated.

**What information can be monitored on the website?**

When registered Technical Support liaisons log into the MicroStrategy Activation website, they can display a list of all Server Installations. This report lists each Server Installation along with the following information for reference:

- Installation ID
- Activation Status
- Contract ID
- Operating System
- Machine CPUs
- Last Update Type
- Last Updated by
- Last Update Date
- Products installed
- Activation and deactivation history

The Installation ID is a unique identifier for Server Installations. This ID is provided along with the Activation Code in the email received when requesting activation.

**Does Server Activation apply to MicroStrategy Reporting Suite?**

Yes. The MicroStrategy Reporting Suite must be activated within 30 days of installation.
Introduction

After installing MicroStrategy, you must complete a few configuration tasks. This chapter addresses the processes used to configure the databases you intend to use in your business intelligence system, as well as an installed MicroStrategy suite of products.

The MicroStrategy platform includes a Tutorial project, which is a sample data warehouse and a demonstration project you can use to learn about the various features that MicroStrategy offers. It is ready to be used and requires no additional configuration tasks. If you want to set up a new system using your own data, you must read this chapter and perform the tasks it describes.

This chapter includes the following information:

- Communicating with databases: ODBC and DSNs, page 144
- Initial MicroStrategy configuration, page 157
- Connecting to a data source, page 192
- Creating a project, page 208
• Configuring your MicroStrategy installation, page 208

If you are configuring MicroStrategy on a UNIX or Linux machine that does not have a GUI, you can perform configuration tasks with command line tools. For steps to perform configuration tasks using command line tools in UNIX and Linux, see Chapter 12, Configuring MicroStrategy Using Command Line Tools.

Communicating with databases: ODBC and DSNs

Establishing communication between MicroStrategy and your databases or other data sources is an essential first step in configuring MicroStrategy products for reporting and analysis of your data. This section explains how MicroStrategy communicates with various data sources and the steps required to set up this communication.

ODBC (Open Database Connectivity) is a standard database access method. ODBC enables a single application to access database data, regardless of the database management system (DBMS) that stores the data. A DBMS is a collection of programs that enables you to store, modify, and extract information from a database.

MicroStrategy Intelligence Server, when used in a three- or four-tier configuration, is the application that uses ODBC to access a DBMS. ODBC drivers translate MicroStrategy Intelligence Server requests into commands that the DBMS understands. MicroStrategy Intelligence Server connects to several databases (at a minimum, the data warehouse and the metadata repository) to do its work.

This section describes the ODBC standard for connecting to databases and creating data source names (DSNs) for the ODBC drivers that are bundled with the MicroStrategy applications.
The diagram below illustrates the three-tier metadata and data warehouse connectivity used in the MicroStrategy system.

The diagram shown above illustrates projects that connect to only one data source. However, MicroStrategy allows connection to multiple data sources in the following ways:

- With MicroStrategy MultiSource Option, a MicroStrategy project can connect to multiple relational data sources. For information on MultiSource Option, see Chapter 9, Optimizing and Maintaining Your Project in the Project Design Guide.

- You can integrate MDX cube sources such as SAP BW, Microsoft Analysis Services, and Hyperion Essbase with your MicroStrategy projects. For information on integrating these MDX cubes sources into MicroStrategy, see the MDX Cube Reporting Guide.
This section provides information and instructions on the following tasks:

- Setting up ODBC, page 146
- Defining DSNs, page 151
- Testing ODBC connectivity, page 155

**Setting up ODBC**

The following information assists you in setting up ODBC between MicroStrategy Intelligence Server and your metadata database and data warehouse.

ODBC is a standard method of communicating with database servers. Intelligence Server uses ODBC to connect to and communicate with all database servers in the system. Specifically, ODBC connects to and transfers data to and from data sources within relational databases.

ODBC permits maximum interoperability—an application can access data in diverse DBMSs through a single framework. A client application uses a database driver and a driver manager to make a connection to the data source. A data source, identified by a data source name, is the database or file accessed by a driver. Data source is another term for a logical database within a database server. A database server can contain multiple logical databases or data sources.

When setting up the ODBC environment, you must create a separate ODBC connection to the data warehouse and metadata repository. This requirement is true even if both databases are accessed through the same DBMS. Further description of these two requirements is provided below:

- A **data warehouse** stores the data that users of the system must analyze to track and respond to business trends, and to facilitate forecasting and planning efforts.
- **Metadata** is a repository whose data associates the tables and columns of a data warehouse with user-defined attributes and facts to enable the mapping of business views, terms, and needs to the underlying database structure. Metadata can reside on the same server as the data warehouse or on a different server. It can be stored in different relational DBMSs.

A successful ODBC connection requires the following information:

- A data source name (DSN) is the name for a pointer used by a client application to find and connect to a data source. A data source is the
database accessed by a driver. The information obtained through a DSN generally includes the host computer name or IP address, instance name, and database name. However, the exact information varies depending on the type of database server.

- **An ODBC driver** is a type of software that translates information between the client application (Intelligence Server) and the database server API. Different database servers speak through different APIs. For example, Informix database servers use Informix Command Line Interface (CLI), Oracle database servers use the Oracle API, Sybase database servers use another API, and so on. For more information on ODBC drivers and how they work with MicroStrategy, see *ODBC drivers, page 148*.

- **A connection string** stores the information required to connect to a database server. A connection string usually includes a DSN, as well as the user ID and password required to log in to the database server. This information varies depending on the particular database server.

- **The ODBC driver manager** coordinates communication between a client application and a database server. The client application tells the driver manager that it needs to connect using a particular connection string. The DSN found in this connection string provides the driver manager with the type of database server to which the application needs access along with the driver to be used. From this information, the driver manager initiates the communication.
The following diagram illustrates the ODBC connection requirements to connect MicroStrategy to a database.

In Windows, MicroStrategy uses the ODBC driver manager provided by Microsoft. In UNIX/Linux environments, MicroStrategy includes its own ODBC driver manager to facilitate the MicroStrategy to database server communication. This MicroStrategy ODBC driver manager has the equivalent functionality of the Microsoft Windows ODBC Driver Manager.

**ODBC drivers**

ODBC drivers are DBMS-specific and must be installed on MicroStrategy Intelligence Server prior to creating the ODBC connection to the warehouse and metadata databases. MicroStrategy embeds and brands DataDirect ODBC drivers in the MicroStrategy platform. These drivers are certified to work with MicroStrategy products. In addition, on Windows, a MicroStrategy installation includes Microsoft SQL Server ODBC from Microsoft’s MDAC component.
The purpose of an ODBC driver is to translate MicroStrategy Intelligence Server requests into commands that the DBMS understands. ODBC drivers submit SQL requests and also return results from the data warehouse to MicroStrategy Intelligence Server. Users of the MicroStrategy platform can employ the MicroStrategy-branded ODBC drivers to connect MicroStrategy products to various DBMSs. For a list of the available ODBC drivers for Windows, Solaris, AIX, HP-UX, and Redhat or SUSE Linux that are certified for Intelligence Server and different DBMS types, see Certified ODBC drivers for MicroStrategy Intelligence Server, page 79.

See the MicroStrategy readme file for details about supported and certified ODBC drivers. To access the MicroStrategy readme files:

- On Windows: From the Start menu, point to Programs, then to MicroStrategy, and then choose ReadMe.
- On UNIX/Linux: From the UNIX/Linux File Manager, browse to INSTALL_PATH, where INSTALL_PATH is the directory that you specified as the install directory during installation. Double-click the ReadMe.htm file.

Note the following:

- Although it is possible to use a non-certified driver, it is strongly recommended that you contact your database vendor to obtain a certified driver if the selected driver is not certified as valid.

### Configuring ODBC with MicroStrategy for Windows

MicroStrategy components require 32-bit drivers to achieve ODBC connectivity. These applications call the driver manager, which in turn calls the drivers. Ideally, all ODBC drivers used on machines connected to data sources should be from the same vendor.

### Default location for ODBC and driver files for Windows

The ODBC driver manager and support libraries are usually installed in the C:\WINDOWS\SYSTEM or C:\WINDOWS\SYSTEM32 directories. Refer to
Communicating with databases: ODBC and DSNs

© 2011 MicroStrategy, Inc.

your third-party documentation for the locations of ODBC support and driver files.

The database-specific ODBC drivers are installed in the locations specified during the installation of the drivers. MicroStrategy-branded drivers are installed in C:\Program Files\Common Files\MicroStrategy. If you installed MicroStrategy on a 64-bit Windows environment, MicroStrategy-branded drivers are installed in C:\Program Files (x86) \Common Files\MicroStrategy.

### Configuring ODBC with MicroStrategy for UNIX and Linux

MicroStrategy components on Solaris, AIX, and Linux require 32-bit drivers to achieve ODBC connectivity. MicroStrategy components on HP-UX require 64-bit drivers to achieve ODBC connectivity. These applications call the driver manager, which in turn calls the drivers. Ideally, all ODBC drivers used on machines connected to data sources should be from the same vendor.

### Default location for ODBC and driver files for UNIX and Linux

The ODBC driver manager and support libraries are usually installed in the following default directories:

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Default Location for ODBC Driver Manager Libraries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solaris</td>
<td>INSTALL_PATH/lib32</td>
</tr>
<tr>
<td>AIX</td>
<td>INSTALL_PATH/lib32</td>
</tr>
<tr>
<td>HP-UX</td>
<td>INSTALL_PATH/lib</td>
</tr>
<tr>
<td>RedHat and SUSE Linux</td>
<td>INSTALL_PATH/lib32</td>
</tr>
</tbody>
</table>

The database-specific ODBC drivers are installed in the locations specified during the installation of the drivers. MicroStrategy-branded ODBC drivers are installed in INSTALL_PATH/lib32 for Solaris, AIX, and Linux, where INSTALL_PATH is the directory you specified as the Install Directory in the Install Wizard. If you install on HP-UX, the MicroStrategy-branded ODBC drivers are installed in INSTALL_PATH/lib.

The MicroStrategy Connectivity Wizard lists only the MicroStrategy-branded ODBC drivers. However, this guide also provides information on how to install drivers from other vendors.
with MicroStrategy. For more information, *Creating DSNs for specific ODBC drivers, page 463.*

**Defining DSNs**

After you install an ODBC driver, you can define one or more data sources for it. The DSN should provide a unique description of the data, for example, `Payroll_Project_Metadata` or `Payroll_Warehouse`.

The DSN is the name for a pointer used by a client application (in this case MicroStrategy) to find and connect to a data source. Multiple DSNs can point to the same data source and one DSN can be used by different applications.

You can define a data source connection with a DSN by using:

- The MicroStrategy Connectivity Wizard—configures connectivity to data sources by creating a DSN that uses a MicroStrategy-branded ODBC driver (see *Creating a DSN for a data source, page 151*).

- The Microsoft ODBC Administrator—creates a DSN for an ODBC driver that is not MicroStrategy-branded (see *Managing ODBC and data sources with Microsoft ODBC Administrator, page 154*).

*Note the following:*

- It is strongly recommended you use the MicroStrategy Connectivity Wizard when creating a new DSN for a MicroStrategy-branded ODBC driver. Use the Microsoft ODBC Administrator only if you intend to use a driver that is not MicroStrategy-branded.

- If you create DSNs using the Microsoft ODBC Administrator, you must create system DSNs. Otherwise, MicroStrategy interfaces will not recognize them.

**Creating a DSN for a data source**

If a DSN does not already exist in your empty metadata repository or the repository installed with MicroStrategy, you can add or create a new one.

The MicroStrategy Connectivity Wizard is a tool designed specifically to configure connectivity to data sources by creating a DSN that uses a MicroStrategy-branded ODBC driver.
To create a DSN

1. If you are creating a DSN:

   - On Windows, perform the following steps:
     - Log in to the system as an administrator.
     - From the Start menu, point to Programs, then MicroStrategy, then Tools, and then select Connectivity Wizard. The Welcome page of the Connectivity Wizard opens.

   - On UNIX or Linux using the Connectivity Wizard interface, perform the following steps:
     - From a UNIX/Linux console window, browse to HOME_PATH, where HOME_PATH is the directory that you specified as the home directory during installation.
     - Browse to the folder bin and type ./mstrconnectwiz, and then press ENTER. The Welcome page of the Connectivity Wizard opens.

   - On UNIX or Linux from the command line, then perform the following steps:
     - From a UNIX/Linux console window, browse to HOME_PATH, where HOME_PATH is the directory that you specified as the home directory during installation.
     - Browse to the folder bin and type ./mstrconnectwiz -h, and then press ENTER.

     This command displays command line operation syntax and examples for different database platforms. Create your command based on the syntax and examples displayed. Once you perform your command, the DSN is created and you can skip the rest of this procedure. For detailed steps on how to use the command line version of this tool, see Creating a DSN for a data source, page 428 in Chapter 12, Configuring MicroStrategy Using Command Line Tools.

2. Click Next. A list of database drivers is displayed. The list available for Windows is different than the list available for UNIX/Linux. For a list of the available ODBC drivers for Windows, Solaris, AIX, HP-UX, and Redhat or SUSE Linux that are certified for Intelligence Server and...
different DBMS types, see *Certified ODBC drivers for MicroStrategy Intelligence Server, page 79.*

For Windows, if you select **Other Relational Databases** and click **Next**, you see a list of all other drivers installed on your machine, excluding the MicroStrategy and Microsoft SQL Server drivers.

3 Select a database driver with which to create a DSN and click **Next**. The Driver Details page opens.

Only a few databases can contain metadata repositories. For details, refer to the MicroStrategy readme. Only DSNs created to connect to these databases can be used to connect to metadata repositories.

4 Enter the information in the appropriate fields for connecting with the selected database driver. The information to enter varies depending on the database platform that you selected. For more information, see *Creating DSNs for specific ODBC drivers, page 463.*

5 Click **Test** to verify the connection. The Test Connection dialog box opens.

6 Enter the **User Name** and **Password** to connect to the database.

7 Click **Connect** to test and verify the connection. If the test is performed successfully, the connection with the database is established. If the test fails, verify the correct connection information with your database administrator and make any required changes to the information you provided in the previous steps.

8 Click **Close**, and then **Finish** to create the new DSN.

If you already have an existing DSN with the same name as the one you provided, a message box appears. You have the following options:

- Select **Yes** to make sure the DSN points to the location you are expecting. This overwrites the existing DSN.
- Select **No** to save the DSN with a different name.

9 Repeat the above steps to create as many DSNs as you require. At a minimum, create one for your metadata and one for your warehouse.
Managing ODBC and data sources with Microsoft ODBC Administrator

The Microsoft ODBC Administrator manages database drivers and data sources on Windows. The Microsoft ODBC Administrator utility creates a log with which to trace calls to data sources and to view, create, and edit DSNs. The utility is available from Microsoft and is usually included with the purchase of an ODBC driver.

Note the following:

- It is strongly recommended that you use the Connectivity Wizard when creating a new DSN for a MicroStrategy-branded ODBC Driver. Use the Microsoft ODBC Administrator only if you intend to use a non-MicroStrategy driver.

- If you choose to create DSNs using the Microsoft ODBC Administrator, they must be system DSNs. Otherwise, MicroStrategy interfaces cannot recognize them.

To create a DSN using the Microsoft ODBC administrator

1. Log in to the machine as an administrator. This gives you the ability to create a system-wide DSN, rather than a user-specific DSN.

2. In most Windows systems you can access the ODBC Data Source Administrator from the Control Panel with the following workflow. From the Windows Start menu, point to Settings, and then choose Control Panel. In the Control Panel dialog box, double-click Administrative Tools, and then double-click Data Sources (ODBC).

   If this workflow does not reflect your Windows system, refer to your third-party documentation for steps to access the ODBC Data Source Administrator tool.

3. Click the System DSN tab. A list displays all the existing system data sources and their associated drivers.

   To view all the installed ODBC drivers, click the Drivers tab.

5 Select the desired driver and click **Finish**. A driver setup dialog box is displayed.

It is recommended that you select a MicroStrategy ODBC driver. These drivers, whose names start with MicroStrategy, were installed when you installed the MicroStrategy application on the computer.

6 Enter the information in the appropriate fields to create a data source for the selected database driver.

The information to enter varies depending on the database platform that you selected, which is discussed in *Creating DSNs for specific ODBC drivers, page 463*.

7 Click **OK** to create a new DSN.

**Testing ODBC connectivity**

ODBC connectivity is one of two layers of connectivity that are listed in the next table, along with the associated connectivity testing programs. Connectivity should be tested from the bottom up—the network layer first and then the ODBC layer.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Test with</th>
</tr>
</thead>
<tbody>
<tr>
<td>ODBC driver</td>
<td>Test ODBC</td>
</tr>
<tr>
<td></td>
<td><strong>mstrtestodbc</strong> or <strong>mstrtodbcx</strong></td>
</tr>
<tr>
<td>Network</td>
<td>Simple Network Layer Testing Tool</td>
</tr>
<tr>
<td>TCP/IP</td>
<td><strong>Ping</strong>, <strong>PING.EXE</strong>, for TCP/IP</td>
</tr>
</tbody>
</table>

The test method described above reflects the situation when the ODBC driver and the database network software are bundled. If they are not bundled, they must be configured and tested separately, using database-specific tools.

**Using the DB Query Tool**

The MicroStrategy DB Query Tool is available in Windows, UNIX, and Linux to test and troubleshoot connectivity to databases, create and execute SQL commands through ODBC, and run scripts.
Prerequisites

Before you use the DB Query Tool, test the network layer with the network layer utility, PING.EXE. Consult your operating system or network system documentation for details.

To use the DB Query Tool

1 To use the DB Query Tool:

- On Windows using the DB Query Tool interface, perform the following step:
  - From the Windows Start menu, point to Programs, then MicroStrategy, then Tools, and then choose DB Query Tool.

- On Windows from the command line, perform the following steps:
  - From the Windows Start menu, select Run. The Run dialog box opens.
  - In the Open drop-down list, type cmd and click OK. A command prompt opens.
  - Type todbcx.exe and press ENTER. Prompts guide you through testing your ODBC connection from the command line and should be used in place of the steps below. For detailed steps on how to use the command line version of this tool, see Testing and examining ODBC connectivity, page 429 in Chapter 12, Configuring MicroStrategy Using Command Line Tools.

- On UNIX or Linux using the DB Query Tool interface, perform the following steps:
  - In a UNIX/Linux console window, browse to HOME_PATH, where HOME_PATH is the directory that you specified as the home directory during installation.
  - Browse to the folder bin and type ./mstrdbquerytool, then press ENTER.

- On UNIX or Linux from the command line, perform the following steps:
  - In a UNIX/Linux console window, browse to HOME_PATH, where HOME_PATH is the directory that you specified as the home directory during installation.
- Browse to the folder bin and type ./mstrtodbcx, then press ENTER. Prompts guide you through testing your ODBC connection from the command line and should be used in place of the steps below. For detailed steps on how to use the command line version of this tool, see Testing and examining ODBC connectivity, page 429 in Chapter 12, Configuring MicroStrategy Using Command Line Tools.

2 From the Session menu, select Open Connection, or click the Connect icon on the toolbar. The GetDSN dialog box opens. The connection interface varies depending on the destination database.

3 Select the DSN for a data source.

4 Enter the appropriate user name and password.

5 Click Connect. After your connection is opened, the connection string is displayed in the MicroStrategy DB Query Tool at the bottom. Your cursor is inserted automatically in the SQL Statement window.

6 In the SQL Statement window, type a SQL query such as:
   
   ```sql
   select count (*) from Table
   where Table is a system-defined table, such as SYSOBJECTS for Microsoft SQL Server or a MicroStrategy-created table such as DSSMDSYSPROP in the MicroStrategy metadata.
   ```

7 From the Queries menu, select Execute Query. A table of data from the database is displayed in the Query Result window.

8 From the Session menu, select Close Connection to close the database connection.

9 From the File menu, select Exit to close the MicroStrategy DB Query Tool.

The DB Query Tool includes many useful features not discussed here. Refer to the DB Query Tool Online Help for details.

---

**Initial MicroStrategy configuration**

The MicroStrategy Configuration Wizard automates much of the configuration process, prompting you only when critical information is
required. With this tool, you can configure the metadata repository, statistics tables, History List tables, MicroStrategy Intelligence Server, and multiple project sources.

If you are configuring MicroStrategy using the Windows operating system, you must have administrative privileges on the computer on which the Intelligence Server is installed, so that you can define the parameters necessary to start the Intelligence Server and to invoke server-definition objects.

You can also configure your MicroStrategy installation using the Configuration Wizard in silent or unattended mode. This allows you to load an existing setup routine to configure your MicroStrategy installation. For information on running the Configuration Wizard with a response file, see Configuring MicroStrategy with a response file, page 183.

Configuration Wizard prerequisites

Before you begin using the Configuration Wizard you should review and complete the following requirements:

- Install the necessary MicroStrategy products. You should have at least MicroStrategy Desktop and MicroStrategy Intelligence Server installed. For information on how to install MicroStrategy on Windows, see Chapter 2, Installing MicroStrategy on Windows. For information on how to install MicroStrategy on other operating systems, see Chapter 3, Installing MicroStrategy on UNIX and Linux.

- Have access to an empty database location certified to house the metadata. This includes creating DSNs for your databases (see Communicating with databases: ODBC and DSNs, page 144). For a list of certified metadata platforms, see the MicroStrategy Readme.

- In a UNIX or Linux environment, the Configuration Wizard must be able to communicate with Intelligence Server over TCP/IP network protocol. To achieve this, the hosts file in the /etc directory must include one entry identifying the local host in the form:

  IP-address local-machine-name

  For example, 123.4.5.6 FakeISmachine. Modifying the hosts file may require an account with root privileges.

- MicroStrategy products must be configured on the machine on which they are installed. You cannot configure them remotely.
Configuring MicroStrategy software

The MicroStrategy Configuration Wizard opens automatically after you install MicroStrategy products and restart your machine.

You can configure a MicroStrategy Web and Intelligence Server (four-tier), Intelligence Server (three-tier), or direct (two-tier) setup for MicroStrategy.

The following figure describes how to configure MicroStrategy to suit an Intelligence Server (three-tier) environment. It also shows how the various components of the MicroStrategy Configuration Wizard, the metadata repository, Intelligence Server, and the project source interact with each other.

A MicroStrategy Web (four-tier) setup involves configuring a web server to communicate between Intelligence Server and MicroStrategy Web or Web Universal. For more information on deploying MicroStrategy Web or Web Universal, see Chapter 6, Deploying MicroStrategy Web and Web Universal.

The figure below describes how to configure MicroStrategy to suit a direct (two-tier) environment. It also shows how the various components of the
MicroStrategy Configuration Wizard, the metadata repository and the project source interact with each other.

![Diagram showing the interaction between Metadata Repository and Project Source](image)

**Step 1**
Create a metadata repository and statistics tables.

**Step 2**
Create a project source and connect to the metadata repository.

It is not recommended to use a direct setup for the production environment.

The procedure below provides the high-level steps to configure MicroStrategy software through the Configuration Wizard.

---

**To configure MicroStrategy through the Configuration Wizard**

1. If you are configuring MicroStrategy on:
   - Windows, then perform the following step:
     - From the **Start** menu, point to **Programs**, then **MicroStrategy**, and then choose **Configuration Wizard**. The Configuration Wizard opens. Continue to the steps provided in *To select a configuration task, page 161*.
   - Windows from the command line, then perform the following steps:
     - From the Windows **Start** menu, select **Run**. The Run dialog box opens.
     - In the **Open** drop-down list, type `cmd` and click **OK**. A command prompt opens.
     - Type `macfgwiz` and press **ENTER**.

This command displays the command line version of the Configuration Wizard. You can configure the connection of a data source to Intelligence Server by creating a response file or using an
existing response file. The command line prompts guide you through configuring the connection of a data source to Intelligence Server by creating a response file or using an existing response file and should be used in place of the steps below. For information on using a response file to configure MicroStrategy, see *Configuring MicroStrategy with a response file, page 183.*

- **UNIX or Linux using the Configuration Wizard interface, then perform the following steps:**
  - From a UNIX/Linux console window, browse to `HOME_PATH`, where `HOME_PATH` is the directory that you specified as the home directory during installation.
  - Browse to the folder `bin` and type `.mstrcfgwiz`, then press ENTER. The Configuration Wizard opens. Continue to the steps provided in *To select a configuration task, page 161.*

- **UNIX or Linux from the command line, then perform the following steps:**
  - From a UNIX/Linux console window, browse to `HOME_PATH`, where `HOME_PATH` is the directory that you specified as the home directory during installation.
  - Browse to the folder `bin` and type `.mstrcfgwiz-editor`, then press ENTER.

This command displays the command line version of the Configuration Wizard. You can configure the connection of a data source to Intelligence Server by creating a response file or using an existing response file. The command line prompts guide you through configuring the connection of a data source to Intelligence Server by creating a response file or using an existing response file and should be used in place of the steps below. For information on using a response file to configure MicroStrategy, see *Configuring MicroStrategy in command line mode, page 411.*

**To select a configuration task**

2 Choose from the following configuration tasks and then click **Next** to begin the selected task.

- **Create Metadata, History List and Statistics Repository Tables:**
  Runs the SQL scripts necessary to create and initialize the metadata repository, History List tables, and statistics tables in the database
location that you select. For steps to complete these configuration tasks, see Metadata, History List, and statistics repository tables, page 162.

- **Configure Intelligence Server**: Creates a new server definition object in the metadata repository that you select. This setup provides Intelligence Server (three-tier) access to all projects that are stored in the repository. This option also allows you to use or delete an existing server definition. For steps to complete these configuration tasks, see MicroStrategy Intelligence Server, page 163.

- **Create Project Sources**: A project source contains the configuration information that each client system requires to access an existing project. It stores the location of the metadata repository and Intelligence Server that is used to run the project. A project source determines how MicroStrategy Desktop, Web, and other client applications access the metadata. For steps to complete these configuration tasks, see Project sources, page 163.

- **Upgrade existing environment to MicroStrategy 9**: You can use the Configuration Wizard to upgrade your MicroStrategy environment and migrate various features to the new version. For all available upgrade and migration options, see the MicroStrategy Upgrade Guide.

The remainder of this chapter describes each configuration option in detail.

After completing these steps, an empty metadata repository is created. To learn how to add projects to your metadata repository, see the MicroStrategy Project Design Guide.

**Metadata, History List, and statistics repository tables**

The metadata consists of tables that contain the definitions for nearly all MicroStrategy objects, including database logins, server definitions, database instances and connections, reports, metrics, facts, and so on. The metadata repository contains information that facilitates the transfer of data among MicroStrategy applications, and between MicroStrategy applications and the data warehouse. MicroStrategy applications use the metadata database to transform user requests into SQL queries. The results of these queries are then translated back as actionable data through MicroStrategy objects such as reports and documents. It is mandatory to have a metadata repository to which Intelligence Server can connect.
History List tables are used to store users’ report and document results for future use. History Lists can be stored on file systems of a server machine. The History List tables provide an alternative option to store History List information in a centralized database. For information on managing History Lists, see the System Administration Guide.

Statistics tables are used to record a variety of statistical information about the usage and performance of a MicroStrategy system. You can set which statistics are recorded in the statistics tables, and then use MicroStrategy Enterprise Manager to analyze the data and locate opportunities for system performance tuning. For information on the statistics tables and Enterprise Manager, refer to the System Administration Guide.

For steps to create metadata, History List, and statistics repositories, see Creating metadata, History List, and statistics repositories, page 164.

MicroStrategy Intelligence Server

MicroStrategy Intelligence Server requires a server definition. A server definition is a MicroStrategy object stored in the metadata repository. It contains information about the configuration of Intelligence Server such as governing settings, which projects should be loaded, which communication protocols should be used, and so on.

Before you can use Intelligence Server, you must use the MicroStrategy Configuration Wizard to define the server definition that Intelligence Server should use.

Many server definitions can exist in the metadata, but you can install only one Intelligence Server on one server machine and Intelligence Server uses only one server definition at a time.

For steps to set up Intelligence Server, see Setting up MicroStrategy Intelligence Server, page 173.

Project sources

Project sources are the highest level objects in the MicroStrategy environment, which represent a connection to a metadata database or a MicroStrategy Intelligence Server. The project source stores the location of the metadata repository or the MicroStrategy Intelligence Server definition that is used to run the project. Through a project source you can create, manipulate, and administer MicroStrategy projects.

For steps to create project sources, see *Creating project sources, page 178*.  

Creating metadata, History List, and statistics repositories

You can create metadata, History List, and statistics repositories (a collection of database tables) using the MicroStrategy Configuration Wizard. Repositories for your metadata, History List, and statistics tables are created in the data source specified by the DSN(s) you connect to.

It is recommended that you create the metadata, History List, and statistics repository tables in different databases to ensure enhanced performance.

You can choose to create metadata, History List, and statistics repositories using a response file with the Configuration Wizard. This lets you provide users with a configuration file to complete the configuration tasks rather than requiring users to step through the Configuration Wizard. Creating and using a response file can be done using the Configuration Wizard interface or a command line tool available for UNIX and Linux. The steps to perform these two configuration options are provided in the sections listed below:

- *Configuring MicroStrategy with a response file, page 183*

As you complete the configuration process, messages may be displayed. For details on system messages displayed during the configuration process, see *Configuration messages, page 173*.

Creating a metadata repository

The metadata repository is a collection of tables that contain the definitions for nearly all MicroStrategy objects including database logins, server definitions, database instances and connections, reports, metrics, facts, and so on. You can create the metadata repository in the database location of your choice. Additionally, a default configuration is created in the metadata tables. This populates the tables with the basic data required for the MicroStrategy metadata, such as the default project folder structure and some basic connection information.

If you are upgrading your metadata from a previous version of MicroStrategy rather than creating a brand new metadata, see the *Upgrade Guide*. 
Prerequisites

- Before you create a metadata repository, you should ensure that you are storing it on a certified database, ODBC driver, and operating system combination. For a list of certified metadata repository environments, see the MicroStrategy Readme.

- A database user account to associate with the metadata tables. This account requires Select, Insert, Update, Create, Drop, and Delete permissions.

To create a metadata repository

1. Open the MicroStrategy Configuration Wizard. To do this, see To configure MicroStrategy through the Configuration Wizard, page 160.


3. Select the Metadata Tables check box and click Next. The Repository Configuration: Metadata tables page opens.

   You can also select to create a History List and a statistics repository immediately after creating a metadata repository. If you create a History List or statistics repository as part of the same configuration routine as creating a new metadata repository, database instances are automatically created for the History List and statistics repositories.

4. From the DSN drop-down list, select the DSN for your metadata repository.

   If a DSN for your metadata repository does not exist, you can click New to open the Connectivity Wizard and create a new DSN. The steps to create a new DSN with the Connectivity Wizard are described in Creating a DSN for a data source, page 151.

   Although it is possible to use the Microsoft Access database for the metadata repository, it is not a suitable metadata repository for a production project. You should not use Microsoft Access for anything other than a proof-of-concept or demonstration type of application.

5. Type a User Name and Password that can connect to the data source.

   The database user that you provide becomes the owner of all metadata tables and objects. The database user is required to have the Select,
Insert, and Update permissions. Intermediate tables are created in the metadata for recursive search queries, which requires Create and Drop permissions as well. Updating the schema requires the Delete permission.

**To specify a metadata table prefix and complete metadata repository creation**

6  Click Advanced. Options to specify a table prefix and a SQL script to create metadata tables are displayed.

7  In the Table Prefix field, you can specify a prefix to be used when metadata tables are created in the database you select. This is an optional configuration. However, you must use different prefixes for your metadata tables and your History List tables if you store them in the same database.

   Most databases use a prefix of two characters. However, you can supply as many letters, numbers, underscores ( _ ), and periods ( . ) as required to support your database prefixes. To determine character limits for a prefix, refer to your third-party database vendor documentation.

8  In the Script field, a SQL script to create metadata tables optimized for your database is selected. If you want to select a different script, click ... to browse to and select a customized script. For more information on the default SQL scripts, see SQL scripts, page 172.

9  Click Next. The next configuration page that opens depends on your configuration scenario:

   •  If your metadata repository does not need to be upgraded and you did not select to create History List or statistics tables, the Summary page opens. You can complete your configuration as described in *To review and save your metadata configuration* below.

   •  If your metadata repository does not need to be upgraded and you selected to configure History List or statistics tables, you are prompted to configure these options as described in *Creating a History List repository, page 167* and *Creating a statistics repository, page 170*.

   •  If your metadata repository needs to be upgraded, cancel this metadata creation process. If you continue with this process of creating metadata tables, your current metadata will be overwritten with a brand new metadata. For information on upgrading your metadata and suite of MicroStrategy projects, refer to the Upgrade Guide.
To review and save your metadata configuration

10 Review the summary information.

You can click Save to save the configuration as a response (.ini) file to configure metadata repositories on other systems or to run silent configurations at a later time. For information on running the Configuration Wizard with a response file, see Configuring MicroStrategy with a response file, page 183.

11 Click Finish to apply the configuration and create the metadata repository. The summary information is updated as the configurations are completed, providing a way to track the progress of the configurations.

Creating a History List repository

A History List repository stores users’ report and document results for future use. History Lists can be stored on file systems of a server machine. The History List tables provide an alternative option to store History List information in a centralized database.

If you create a History List repository as part of the same configuration routine to create a metadata repository, a database instance is automatically created for the History List repository. If you create the History List repository separately or you create it for an existing metadata repository, you must create a database instance for the History List repository. For information on creating a database instance, see Creating a database instance, page 197.

If you are upgrading your History List repository from a previous version of MicroStrategy rather than creating a brand new History List repository, see the Upgrade Guide.

Prerequisites

• Before you create a History List repository, you should ensure that you are storing it on a certified database, ODBC driver, and operating system combination. For a list of certified History List repository environments, see the MicroStrategy Readme.

• A database user account to associate with the History List tables. The database user is required to have the Select, Create, Insert, and Drop permissions.
• The steps below are specific to creating a History List repository. If you also select to create a metadata repository, you must first complete the steps described in *Creating a metadata repository, page 164.*

---

**To create a History List repository**

1. Open the MicroStrategy Configuration Wizard. To do this, see *To configure MicroStrategy through the Configuration Wizard, page 160.*

2. Select **Metadata, History List and Statistics Repository Tables** and click **Next**. The Repository Configuration: Repository Types page opens.

3. Select the **History List Tables** check box and click **Next**. The Repository Configuration: History List tables page opens.

4. From the **DSN** drop-down list, select the DSN for your History List repository.

   If a DSN for your History List repository does not exist, you can select **New** to open the Connectivity Wizard and create a new DSN. The steps to create a new DSN with the Connectivity Wizard are described in *Creating a DSN for a data source, page 151.*

5. Type a **User Name** and **Password** that can connect to the data source.

   The database user you provide becomes the owner of all History List tables and objects. The database user is required to have the Select, Create, Insert, and Drop permissions.

   **To specify a History List table prefix and complete History List repository creation**

6. Click **Advanced**. Options to specify a table prefix and a SQL script to create History List tables are displayed.

7. In the **Table Prefix** field, you can specify a prefix to be used when History List tables are created in the database you select. This is an optional configuration. However, you must use different prefixes for your metadata tables and your History List tables if you store them in the same database.

   Most databases use a prefix of two characters. However, you can supply as many letters, numbers, underscores (_), and periods (.) as required to support your database prefixes. To determine character limits for a prefix, refer to your third-party database vendor documentation.
If you use a table prefix for your History List tables, you must also define this table prefix when you create a database instance to connect to the History List tables. For information on creating a database instance, see *Creating a database instance, page 197.*

8 In the **Script** field, a SQL script to create History List tables optimized for your database is selected. If you want to specify a different script, click ... to browse to and select a customized script. For more information on the default SQL scripts, see *SQL scripts, page 172.*

9 Click **Next**. The next configuration page that opens depends on your configuration scenario:

- If you did not select to create statistics tables, the Summary page opens. You can complete your configuration as described in *To review and save your History List configuration* below.

- If you selected to configure statistics tables, you are prompted to configure these options as described in *Creating a statistics repository, page 170.*

**To review and save your History List configuration**

10 Review the summary information.

You can click **Save** to save the configuration as a response (.ini) file to configure History List repositories on other systems or to run silent configurations at a later time. For information on running the Configuration Wizard with a response file, see *Configuring MicroStrategy with a response file, page 183.*

11 Click **Finish** to apply the configuration and create the History List repository. The summary information is updated as the configurations are completed, providing a way to track the progress of the configurations.

Once you are finished configuring Intelligence Server and your project sources, a database instance to connect a project to a History List repository must be created. A database instance is automatically created for the History List repository if you created the History List repository as part of the same configuration routine to create a metadata repository. For information on creating a database instance, see *Creating a database instance, page 197.*
Creating a statistics repository

A statistics repository is a collection of database tables used to maintain and monitor system activity and performance. You can run MicroStrategy Enterprise Manager against the statistical information to analyze and interpret the statistics.

For a detailed description of the statistics tables used in the Enterprise Manager Statistics database, the fields that each table contains, and the data types associated with each field for MicroStrategy Intelligence Server, refer to Chapter 6, Enterprise Manager Data Model and Object Definitions in the Supplemental Reference for System Administration Guide.

For details on how to configure projects to log statistics, refer to Chapter 6, Monitoring System Usage in the System Administration Guide.

If you create a statistics repository as part of the same configuration routine to create a metadata repository, a database instance is automatically created for the statistics repository. If you create the statistics repository separately or you create it for an existing metadata repository, you must create a database instance for the statistics repository. For information on creating a database instance, see Creating a database instance, page 197.

If you are upgrading your statistics repository from a previous version of MicroStrategy rather than creating a brand new statistics repository, see the Upgrade Guide.

Prerequisites

• Before you create a statistics repository, you should ensure that you are storing it on a certified database, ODBC driver, and operating system combination. For a list of certified statistics repository environments, see the MicroStrategy Readme.

• A database user account to associate with the statistics tables. The database user is required to have the Select, Create, Insert, and Drop permissions.

• The steps below are specific to creating a statistics repository. If you also select to create a metadata repository or History List repository, you must first complete the steps described in Creating a metadata repository, page 164 or Creating a History List repository, page 167, respectively.
To create a statistics repository

1. Open the MicroStrategy Configuration Wizard. To do this, see *To configure MicroStrategy through the Configuration Wizard, page 160*.

2. Select **Metadata, History List and Statistics Repository Tables** and click **Next**. The Repository Configuration: Repository Types page opens.

3. Select the **Statistics Tables** check box and click **Next**. The Repository Configuration: Statistics Tables page opens.

4. From the **DSN** drop-down list, select the DSN for your statistics repository.

   If a DSN for your statistics repository does not exist, you can select **New** to open the Connectivity Wizard and create a new DSN. The steps to create a new DSN with the Connectivity Wizard are described in the section *Creating a DSN for a data source, page 151*.

5. Type a **User Name** and **Password** that can connect to the data source.

   The database user you provide becomes the owner of all statistics tables and objects. The database user is required to have the Select, Create, Insert, and Drop permissions.

   **To specify a statistics table prefix and complete statistics repository creation**

6. Click **Advanced**. Options to specify a table prefix and a SQL script to create statistics tables are displayed.

7. In the **Table Prefix** field, you can specify a prefix to be used when statistics tables are created in the database you select. This is an optional configuration.

   Most databases use a prefix of two characters. However, you can supply as many letters, numbers, underscores (_), and periods (.) as required to support your database prefixes. To determine character limits for a prefix, refer to your third-party database vendor documentation.

   If you use a table prefix for your statistics tables, you must also define this table prefix when you create a database instance to connect to the statistics tables. For information on creating a database instance, see *Creating a database instance, page 197*. 
8 In the **Script** field, a SQL script to create statistics tables optimized for your database is selected. If you want to specify a different script, click ... to browse to and select a customized script. For more information on the default SQL scripts, see *SQL scripts, page 172*.

9 Click **Next**. The Summary page opens.

10 Review the summary information.

   You can click **Save** to save the configuration as a response (.ini) file to configure statistics repositories on other systems or to run silent configurations at a later time. For information on running the Configuration Wizard with a response file, see *Configuring MicroStrategy with a response file, page 183*.

11 Click **Finish** to apply the configuration and create the statistics repository. The summary information is updated as the configurations are completed, providing a way to track the progress of the configurations.

Once you are finished configuring Intelligence Server and your project sources, a database instance to connect a project to a statistics repository must be created. A database instance is automatically created for the statistics repository if you created the statistics repository as part of the same configuration routine to create a metadata repository. For information on creating a database instance, see *Creating a database instance, page 197*.

### SQL scripts

MicroStrategy has database-specific **SQL** scripts for creating metadata, History List, and statistics tables. The scripts for each certified database platform are shipped with the product. The MicroStrategy Configuration Wizard automatically selects a default script based on your ODBC driver’s database platform.

By default, all the scripts reside in the directory where you installed MicroStrategy and are identified by the .sql extension. It is highly recommended that no edits be performed on these scripts, except on rare occasions and only by skilled database personnel.
**Configuration messages**

Depending on the selected ODBC database, different messages might be displayed prompting you to complete the configuration successfully. Two examples are described below:

- **Metadata tables already exist at this location. Would you like to recreate them? (This will drop all existing information in the Metadata)**

  This message is displayed if the Configuration Wizard detects an existing metadata repository in the database location you specified. If you continue, all information in the existing metadata repository is overwritten.

- **No Metadata Tables were found at this location, do you wish to create them now?**

  This message is displayed if there is no existing metadata repository and you have not chosen to create one.

**Setting up MicroStrategy Intelligence Server**

You use the Configuration Wizard to create, use, or delete server definitions that are stored in the metadata repository. The following list describes these options in detail:

- **Create**: When you create a new server definition in the metadata repository of your choice, all its parameters use the default settings. Creating a new server definition creates an entry in the machine's configuration file. This entry points to the metadata and is used by MicroStrategy Intelligence Server at startup. You can modify a server definition using the MicroStrategy Intelligence Server Configuration Editor. For information on the Intelligence Server Configuration Editor, refer to the *MicroStrategy System Administration Guide*.

- **Use**: When you use an existing server definition different from the current server definition, you are changing the machine's configuration information to point to a different server definition, which can be in an entirely different metadata.
• **Delete:** When you delete a server definition, you are deleting the server definition object from the metadata repository, but not from the Intelligence Server software that you installed.

If you create a new server definition or use an existing one, you can load the projects to use with the server definition from within the Configuration Wizard.

If the password to the metadata has changed in the database, use the Server Configuration: Metadata Connection page in the Configuration Wizard to update your password.

A procedure to use the Configuration Wizard to define server definitions is provided in *To set up MicroStrategy Intelligence Server* below.

You can choose to configure the server definition, project source names, and the metadata and statistics repositories using a response file with the Configuration Wizard. This lets you provide users with a configuration file to complete the configuration tasks rather than requiring users to step through the Configuration Wizard. Creating and using a response file can be done using the Configuration Wizard interface or a command line tool available for UNIX and Linux. The steps to perform these two configuration options are provided in the sections listed below:

• **Configuring MicroStrategy with a response file, page 183**

• **Configuring MicroStrategy with a response.ini file, page 430** in *Chapter 12, Configuring MicroStrategy Using Command Line Tools*

---

**To set up MicroStrategy Intelligence Server**

1. Open the MicroStrategy Configuration Wizard. To do this, see *To configure MicroStrategy through the Configuration Wizard, page 160*.

2. Select **MicroStrategy Intelligence Server**, and click **Next**. The Server Configuration: Metadata Connection page opens.

3. From the DSN drop-down list, select a DSN for the data source that stores the metadata and specify a **User Name** and **Password**.

You can also use the options listed below:

• **New** to create a new DSN (see *Creating a DSN for a data source, page 151*)
• **Advanced** if you want to specify a metadata table prefix, which is an identifier stored in the project metadata associated with a table or tables and is used to generate SQL.

Although it is possible to use the Microsoft Access database for the metadata repository, it is not a suitable metadata repository for a production project. You should not use Microsoft Access for anything other than a proof-of-concept or demonstration type of application.

4 Click **Next**. If a message is displayed that your metadata is not up to date with the most recent version of MicroStrategy, you must upgrade your metadata to take advantage of the new features available in the most recent version of MicroStrategy. You can upgrade your MicroStrategy metadata as described in the *MicroStrategy Upgrade Guide*.

5 In the Server Configuration: MicroStrategy Authentication page, specify the MicroStrategy administrator’s **User Name** and **Password**. By default, the user name is Administrator and it has no password. If you are setting up Intelligence Server for the first time, use the default user name and password.

For security reasons, you should change the Administrator user name and password as soon as possible after you initially configure the system. Refer to the *System Administration Guide* for details about passwords and other user management information.

6 Click **Next**. The Server Configuration: Server Definitions page opens.

**To create, use, or delete a server definition**

7 You can create a new server definition, use an existing server definition, or delete a current server definition. You can perform one of the following tasks:

- To create a server definition, select **Create New Server Definition**. In the **Name** field, type a name to identify the server definition. Select the **Use as the active server definition** check box to define Intelligence Server to use the new server definition when Intelligence Server starts.

Click **Next**. The Server Configuration: Settings page opens, described in *To define the Intelligence Server port number and other settings* below.
To use an existing server definition, select **Use the selected Server Definition as active**. From the Existing Server Definitions pane, select a server definition to use. Click **Next**. The Server Configuration: Settings page opens, described in *To define the Intelligence Server port number and other settings* below.

To delete an existing server definition, select **Delete Selected Server Definition**. From the Existing Server Definitions pane, select a server definition to delete. Click **Next**. The Summary page opens, described in *To complete Intelligence Server configurations* below.

**To define the Intelligence Server port number and other settings**

8 If you select to create a server definition or use an existing server definition, you can define the Intelligence Server port number and other settings, as described below:

- **Port number**: You can use the default *port number* or specify another port number. The port number is how a server process identifies itself on the machine on which it is running. If the port number is used by another process, such as in a shared environment, specify an available port number. For instructions on how to find an available port number, see *Port number is in use, page 531*.

- **Register Intelligence Server as a Service**: This option is only available if you are configuring Intelligence Server on a UNIX or Linux machine, and you have root access and permissions to the UNIX or Linux machine that Intelligence Server is installed on. Select this check box to register Intelligence Server as a service.

In Windows, Intelligence Server is automatically registered as a service upon completion of the configuration process.

Running the Configuration Wizard again and clearing this check box does not unregister Intelligence Server as a service. To unregister Intelligence Server on UNIX or Linux, you must stop the service, and then use the `mstrctl` command line tool. The syntax is `mstrctl -s IntelligenceServer us`, where `IntelligenceServer` is the name of a server definition. For information on starting, stopping, and registering Intelligence Server as a service, see the *System Administration Guide*.

- **Projects to load at startup**: This pane displays all the projects that are in the metadata repository. You can select projects to use with the server definition that you have chosen. The projects that you select are loaded on the server at startup.
• **Start Intelligence Server when finished**: Select this check box to have Intelligence Server start once you complete this configuration.

If you use Windows NT authentication with SQL Server, you must type the Windows NT account user name and password in the Service Manager to successfully start Intelligence Server. For information on the Service Manager, see the *System Administration Guide*.

• **Identify missing DSNs**: Select this check box to verify that all DSNs, which are used for database instances created in MicroStrategy, are locally available. This helps to ensure that your database instances in MicroStrategy can connect successfully to their associated data sources.

By default, this check box is cleared, which means the availability of all local DSNs used in database instances is not verified. While this may mean that all DSNs used in database instances are not available, it can save system resources required for the Intelligence Server configuration process.

9 Click **Next**. The Summary page opens.

**To complete Intelligence Server configurations**

10 Review the summary information.

You can click **Save** to save the configuration as a response (.ini) file to configure Intelligence Server on other systems or to run silent configurations at a later time. For information on running the Configuration Wizard with a response file, see Configuring MicroStrategy with a response file, page 183.

11 Click **Finish** to apply the Intelligence Server configuration. The summary information is updated as the configurations are completed, providing a way to track the progress of the configurations.

If you created a new server definition, it is displayed in the list of existing server definitions for that metadata.

If you assigned an existing server definition to Intelligence Server and the existing project source uses this Intelligence Server, a related message is displayed.
Starting, stopping, and restarting the server

The primary function of the Service Manager is to provide a way to start and stop Intelligence Server. For information on the Service Manager, refer to the System Administration Guide.

Creating project sources

When you create a metadata repository, by default it creates a server (three-tier) project source. You can use the Project Sources option in the MicroStrategy Configuration Wizard if you need to create multiple project sources or a direct (two-tier) project source.

The different types of project sources are:

- Direct project sources that connect directly to the metadata through ODBC. You cannot create a direct project source on UNIX or Linux.
- Server project sources that connect to the metadata through an Intelligence Server.

Prerequisites

- For Windows, the Project Source option is available only if the Desktop product is installed on the machine.

You can choose to create project sources using a response file with the Configuration Wizard. This lets you provide users with a configuration file to complete the configuration tasks rather than requiring users to step through the Configuration Wizard. Creating and using a response file can be done using the Configuration Wizard interface or a command line tool available for UNIX and Linux. The steps to perform these two configuration options are provided in the sections listed below:

- Configuring MicroStrategy with a response file, page 183
**Direct (two-tier)**

A direct project source is used to connect directly to the metadata repository using ODBC. A direct project source connection does not allow you to access MicroStrategy Web, run Report Services documents, or use any of the other MicroStrategy features that are provided through Intelligence Server.

You cannot create a direct project source on UNIX or Linux.

---

**To create a direct project source**

1. Open the MicroStrategy Configuration Wizard. To do this, see *To configure MicroStrategy through the Configuration Wizard, page 160.*

2. Select **Project Sources**, and click **Next**. The Project Source Creation: Name page opens.

3. In the **Project Source Name** field, type a name for the project source.

4. Under **Connection Type**, select **Direct (2-tier)**, and click **Next**. The Project Source Creation: Metadata Location page opens.

5. From the **DSN** drop-down list, select a DSN for the data source that stores the metadata and specify a **User Name** and **Password**.

   You can also click **New** to create a new DSN (see *Creating a DSN for a data source, page 151*) and click **Advanced** to specify a metadata table prefix if necessary.

6. Click **Next**. The Project Source Creation: Authentication page opens.

7. Select the authentication mode for the project source. For information on the available authentication modes, see the *Authentication modes, page 181.*

8. Click **Next**. The Summary page opens.

9. Review the summary information.

You can click **Save** to save the configuration as a response (.ini) file to configure a direct project source on other systems or to run silent configurations at a later time. For information on running the Configuration Wizard with a response file, see *Configuring MicroStrategy with a response file, page 183.*
10 Click **Finish** to create the project source. The summary information is updated as the configurations are completed, providing a way to track the progress of the configurations.

**MicroStrategy Intelligence Server (three-tier)**

A server (three-tier) project source is used to connect to the metadata using the MicroStrategy Intelligence Server. A server project source connection allows you to access MicroStrategy Web, run Report Services documents, and use all of the other MicroStrategy features that are provided through Intelligence Server.

When you create a metadata repository, by default it creates a server (three-tier) project source.

---

**To create a MicroStrategy Intelligence Server (three-tier) project source**

1. Open the MicroStrategy Configuration Wizard. To do this, see *To configure MicroStrategy through the Configuration Wizard, page 160.*

2. Select **Project Sources** and click **Next**. The Project Source Creation: Name page opens.

3. In the **Project Source Name** field, type a name for the project source.

4. Under **Connection Type**, select **MicroStrategy Intelligence Server (3-tier)**, and click **Next**. The Project Source Creation: Metadata Location page opens.

5. In the **MicroStrategy Intelligence Server Machine Name** drop-down list, select the Intelligence Server to connect to.

6. In the **Port Number used by MicroStrategy Intelligence Server** field, type the port number for the Intelligence Server to connect to.

   The port number is how the Intelligence Server process identifies itself on the server on which it is running. The default port number for Intelligence Server is 34952. If you use a non-default port number, this number must be provided while connecting through MicroStrategy Desktop.

7. Select the **Connection times out after (mins) check box** to define and enforce a connection time out for inactive users connected to a project.
source. In the field below, type a numerical value (in minutes) for the amount of inactivity that is allowed before a user is automatically disconnected from a project source. If this check box is cleared, users are not disconnected from project sources due to inactivity.

8 Click **Next**. The Project Source Creation: Authentication page opens.

9 Select the authentication mode for the project source. For information on the available authentication modes, see the *Authentication modes*, page 181.

10 Click **Next**. The Summary page opens.

11 Review the summary information.

You can click **Save** to save the configuration as a response (.ini) file to configure a server project source on other systems or to run silent configurations at a later time. For information on running the Configuration Wizard with a response file, see *Configuring MicroStrategy with a response file*, page 183.

12 Click **Finish** to create the project source. The summary information is updated as the configurations are completed, providing a way to track the progress of the configurations.

**Authentication modes**

Authentication is the process through which the system identifies the user. Several authentication modes are supported for MicroStrategy project sources. They vary primarily by the system that verifies and accepts the login/password credentials provided by the user.

Some authentication modes require a server project source (three-tier). Therefore, if you are creating a direct project source (two-tier) some of the authentication options listed below cannot be used:

- **Network login ID: Windows authentication**, page 182
- **Login ID and password entered by the user: Standard authentication**, page 182
- **Guest account: Anonymous authentication**, page 182
- **LDAP authentication**, page 183
• **Login ID and password entered by the user for the warehouse:**
  *Database authentication, page 183*

• **Integrated authentication, page 183**

For information on the benefits of the various authentication modes and other authentication topics, see the *System Administration Guide.*

**Network login ID: Windows authentication**

To use Windows authentication, you must create users in the MicroStrategy environment and then link them to Windows users. If you use Windows as your network operation system and your users are already defined in the Windows directory, your users can access the MicroStrategy application without having to enter a login ID and password.

**Login ID and password entered by the user: Standard authentication**

When using standard authentication, the MicroStrategy Intelligence Server is the authentication authority. Intelligence Server verifies and accepts the login and password provided by the user. This information is stored in the metadata repository. When a project source is configured to use standard authentication, users must enter a valid login ID and password combination before they can access the project source. Each user has a unique login/password and can be identified in the MicroStrategy application uniquely. By default, all users connect to the data warehouse using one RDBMS login ID, although you can change this using connection mapping. For information on configuring connection mapping, see the *System Administration Guide.*

**Guest account: Anonymous authentication**

When using anonymous authentication, users log in as "Guest" and do not need to provide a password. By default, guest users can access the project, browse objects, run and manipulate reports, but they cannot create their own objects or schedule report executions. However, you determine what the Guest user can and cannot do by modifying the Public user group. Guest users inherit their privileges from the Public group; they are not part of the Everyone group.
LDAP authentication

Lightweight Directory Access Protocol (LDAP) authentication identifies users within a repository of users stored in an LDAP server (such as Novell Directory Services). If you use an LDAP directory to centrally manage users in your environment, you may want to use LDAP authentication. Group membership can be maintained in the LDAP directory without having to also be defined in the MicroStrategy Intelligence Server. When using LDAP authentication, LDAP users or groups are linked to users or groups in the MicroStrategy environment.

Login ID and password entered by the user for the warehouse: Database authentication

This mode of database authentication identifies users using a login ID and password stored in the data warehouse database. Under this mode of authentication, a warehouse database is associated with each project. When users log in to a project source, they are logging in to the Intelligence Server. Use database authentication if you want the data warehouse RDBMS to be the authority for identifying users and you do not want to maintain user credentials in the Intelligence Server as well as the RDBMS.

Integrated authentication

Integrated authentication enables a Windows user to log in once to their Windows machine. The user does not need to log in again separately to MicroStrategy Desktop or MicroStrategy Web. This type of authentication uses Kerberos to validate a user’s credentials.

Configuring MicroStrategy with a response file

The Configuration Wizard walks you through the process of setting up the environment for the MicroStrategy products installed in your system. You can also configure server definitions, project source names, and the metadata, History List, and statistics repositories using a response file with the Configuration Wizard. This enables you to provide a configuration file to users to complete the configuration tasks, rather than requiring users to step through the Configuration Wizard. This can be done to configure a MicroStrategy installation on Windows, UNIX, and Linux.
Creating a response file

It is recommended that you always create the response file through the graphical interface of the Configuration Wizard, as described in the procedure To create a response file in this section. This ensures that all applicable options are included in the response file with valid values.

However, you can also create and use a response file with the Configuration Wizard in command line mode on UNIX and Linux machines. For steps to create and use a response file as well as perform other configurations using command line tools in UNIX and Linux, see the Configuring MicroStrategy with a response.ini file, page 430 section in Chapter 12, Configuring MicroStrategy Using Command Line Tools.

To create a response file

1. Open the MicroStrategy Configuration Wizard. To do this, see To configure MicroStrategy through the Configuration Wizard, page 160.

2. Any configuration tasks you complete with the Configuration Wizard can be saved to a response file. For steps to complete various configurations tasks with the Configuration Wizard, see the sections listed below:
   - Creating metadata, History List, and statistics repositories, page 164
   - Setting up MicroStrategy Intelligence Server, page 173
   - Creating project sources, page 178

3. Once you reach the Summary page for a configuration, click Save. The Save dialog box opens.

4. Specify a name and location to save the response file, and click Save. You are returned to the Summary page.

5. To also perform the configuration task, click Finish. The summary information is updated as the configurations are completed, providing a way to track the progress of the configurations.

Steps to use a response file to configure MicroStrategy are covered in Using a response file to configure MicroStrategy installations below.
You can modify a response file with a text editor to make configuration changes such as entering different user login and password information. For information on the parameters and options available in response files, see Response configuration parameters and options, page 186.

Using a response file to configure MicroStrategy installations

Rather than stepping through each page of the Configuration Wizard, you can configure MicroStrategy using a response file. You have the following options to use a response file to configure MicroStrategy:

- **To use a response file with the Configuration Wizard, page 185**: This covers the standard procedure of running a response file with the Configuration Wizard interface.

- **To use a response file through the Windows command line, page 186**: This covers the procedure of running a response file from the Windows command line. This enables users to run the file without using any graphical user interfaces.

If you are configuring a MicroStrategy installation on UNIX or Linux, you can use a command line version of the Configuration Wizard to create and use a response file. For steps to create and use a response file as well as perform other configurations using command line tools in UNIX and Linux, see the Configuring MicroStrategy with a response.ini file, page 430 section in Chapter 12, Configuring MicroStrategy Using Command Line Tools.

To use a response file with the Configuration Wizard

1. Open the MicroStrategy Configuration Wizard. To do this, see To configure MicroStrategy through the Configuration Wizard, page 160.

2. Click **Load**. The Open dialog box displays.

3. Browse to the path where the response file is saved and click **Open**. The Summary page opens.

4. An overview of all of the configuration tasks performed by the response file is displayed. Review the configuration tasks and click **Finish** to perform the configuration. The summary information is updated as the configurations are completed, providing a way to track the progress of the configurations.
To use a response file through the Windows command line

The steps below are specific to configuring MicroStrategy installed on Windows. For steps to create and use a response file as well as perform other configurations using command line tools in UNIX and Linux, see Chapter 12, Configuring MicroStrategy Using Command Line Tools.

1. Type the following command in the Windows command line:

   \macfgwiz.exe -r "Path\response.ini"

   Where Path\ is the fully qualified path to the response file. For example, the common location of a response file is:

   C:\Program Files\Common Files\MicroStrategy\RESPONSE.INI

2. If an error message is displayed, check the path and name you supplied for the response file and make any required changes. Repeat the previous step to execute the configuration.

Response configuration parameters and options

It is recommended that you always create the response file through the GUI mode of the Configuration Wizard. However, you can also modify a response file with a text editor to make minor changes such as entering different user login and password information.

The file must be saved with ANSI encoding.

The response file for configuring MicroStrategy is divided into three areas of configuration, which are described in the sections below:

- Creating metadata, History List, and statistics repositories, page 186
- Setting up MicroStrategy Intelligence Server, page 189
- Creating and configuring project sources, page 191

Creating metadata, History List, and statistics repositories

The response file parameters within the [Repository] section define how metadata, History List, and statistics repositories are created. The table
below lists the available parameters and the functionality of available options for each parameter.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Repository]</td>
<td>This section configures the metadata repository and statistics tables. You can have more than one [Repository] section. Additional repository sections can be included as [Repository1], [Repository2], and so on.</td>
</tr>
<tr>
<td>Repository=</td>
<td>Defines whether a metadata, History List, and statistics repositories are configured, as determined by the following values:                                                                                     • 1: Configures metadata, History List, and statistics repositories.</td>
</tr>
<tr>
<td></td>
<td>• 0: Does not configure metadata, History List, and statistics repositories.</td>
</tr>
<tr>
<td>CreateMDTables=</td>
<td>Defines whether metadata tables are created in a metadata repository, as described below:                                                                                                                        • 1: Creates metadata tables in the metadata repository and creates a default configuration</td>
</tr>
<tr>
<td></td>
<td>• 0: Does not create metadata tables in a metadata repository.</td>
</tr>
<tr>
<td>CreateHistListTables=</td>
<td>Defines whether a History List repository is created, as determined by the following values:                                                                                                        • 1: Creates a History List repository.</td>
</tr>
<tr>
<td></td>
<td>• 0: Does not create a History List repository.</td>
</tr>
<tr>
<td>CreateStatTables=</td>
<td>Defines whether a statistics repository is created, as determined by the following values:                                                                                                                        • 1: Creates a statistics repository.</td>
</tr>
<tr>
<td></td>
<td>• 0: Does not create a statistics repository.</td>
</tr>
<tr>
<td>MetadataPath=</td>
<td>Locates the SQL scripts for creating the metadata tables. Example paths to SQL scripts in different environments are listed below:                                                                                       • 32-bit Windows environment: C:\Program Files\Common Files\MicroStrategy\MD8SQL8.sql.</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environment: C:\Program Files (x86)\Common Files\MicroStrategy\MD8SQL8.sql.</td>
</tr>
<tr>
<td></td>
<td>• UNIX/Linux: /INTELLIGENCE_SERVER_INSTALL_PATH/mdsql.sql.</td>
</tr>
<tr>
<td>HistoryListPath=</td>
<td>Locates the SQL scripts for creating the History List repository. Example paths to SQL scripts in different environments are listed below:                                                                                                                                  • 32-bit Windows environment: C:\Program Files\Common Files\MicroStrategy\content_server_db_Oracle.sql.</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environment: C:\Program Files (x86)\Common Files\MicroStrategy\content_server_db_Oracle.sql.</td>
</tr>
<tr>
<td></td>
<td>• UNIX/Linux: /INTELLIGENCE_SERVER_INSTALL_PATH/content_server_db_Oracle.sql.</td>
</tr>
<tr>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| StatisticsPath=  | Locates the SQL scripts for creating the statistics repository. Example paths to SQL scripts in different environments are listed below:  
                      • 32-bit Windows environment: C:\Program Files\Common Files\MicroStrategy\statistics_SQLServer.SQL  
                      • 64-bit Windows environment: C:\Program Files (x86)\Common Files\MicroStrategy\statistics_SQLServer.SQL  
                      • UNIX/Linux: /INTELLIGENCE_SERVER_INSTALL_PATH/statistics_DB2.sql.                                                                                          |
| DSNName=         | Defines the Data Source Name for configuring a metadata repository in the ODBC database.                                                                                                                |
| UserName=        | Defines the user name to log in to the database containing the metadata repository.                                                                                                                     |
| UserPwd=         | Defines the password to log in to the database containing the metadata repository.                                                                                                                      |
| DSNNameHist=     | Defines the Data Source Name for configuring the History List repository in the ODBC database.                                                                                                           |
| UserNameHist=    | Defines the user name to log in to the database for configuring the History List repository.                                                                                                            |
| UserPwdHist=     | Defines the password to log in to the database for configuring the History List repository.                                                                                                              |
| DSNNameStats=    | Defines the Data Source Name for configuring the statistics repository in the ODBC database.                                                                                                               |
| UserNameStats=   | Defines the user name to log in to the database for configuring the statistics repository.                                                                                                                |
| UserPwdStats=    | Defines the password to log in to the database for configuring the statistics repository.                                                                                                                |
| EncryptPassword= | Defines whether the password is encrypted in the response file, as determined by the following values:  
                      • 0: The password is not encrypted in the response file, which enables you to modify the password in the response file later using a text editor. You can then distribute the response file to multiple users with various login and password credentials. However, be aware that this can compromise your database security if you do not remove the password from the response file before distributing it.  
                      • 1: Encrypts the password in the response file, which ensures that your password is secure. This is the default behavior.                                                               |
| DBName=          | Defines the database name to create tables in DB2 z/OS. This option should only be used when connecting to a DB2 z/OS database.                                                                              |
| TBName=          | Defines the tablespace name to be created in the database. This option should only be used when connecting to a DB2 z/OS database.                                                                       |
Setting up MicroStrategy Intelligence Server

The response file parameters within the `[Server]` section configures an Intelligence Server definition. The table below lists the available parameters and the functionality of available options for each parameter.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Server]</td>
<td>In this section you can configure the Intelligence Server definition. You can have more than one <code>[Server]</code> section. Additional server sections can be included as <code>[Server1]</code>, <code>[Server2]</code>, and so on.</td>
<td></td>
</tr>
<tr>
<td>Server=</td>
<td>Defines whether MicroStrategy Intelligence Server is configured, as determined by the following values:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 1: Configures MicroStrategy Intelligence Server</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 0: Does not configure MicroStrategy Intelligence Server</td>
<td></td>
</tr>
<tr>
<td>Action=</td>
<td>Defines whether a server definition is created, used, or deleted, as determined by the following values:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 1: Creates a new server definition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 2: Uses an existing server definition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 3: Deletes an existing server definition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 4: Creates a new server definition and uses it as the default</td>
<td></td>
</tr>
<tr>
<td>InstanceName=</td>
<td>Defines the name of the Intelligence Server instance.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If you select to delete Intelligence Server instances, you can delete multiple instances by listing multiple instance names, separating each name with the \ character. For example,</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>InstanceName=ServerInstance1\ServerInstance2</code>.</td>
<td></td>
</tr>
<tr>
<td>ProjectsToRegister=</td>
<td>Defines projects to be loaded when Intelligence Server is started. You can select to load multiple projects, separating projects by the \ character. For example,</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>ProjectsToRegister=Project1\Project2</code>.</td>
<td></td>
</tr>
<tr>
<td>ProjectsToUnRegister=</td>
<td>Defines projects to not be loaded when Intelligence Server is started. You can select to not load multiple projects, separating projects by the \ character. For example,</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>ProjectsToUnRegister=Project1\Project2</code>.</td>
<td></td>
</tr>
<tr>
<td>DSName=</td>
<td>Defines the data source name for configuring the MicroStrategy Intelligence Server. This is the data source that stores the metadata.</td>
<td></td>
</tr>
<tr>
<td>DSNUser=</td>
<td>Defines the user name to log in to the metadata database.</td>
<td></td>
</tr>
<tr>
<td>DSNpwd=</td>
<td>Defines the password to log in to the metadata database.</td>
<td></td>
</tr>
<tr>
<td>Options</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>EncryptPassword=</td>
<td>Defines whether the password is encrypted in the response file, as determined by the following values:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 0: The password is not encrypted in the response file, which enables you to modify the password in the response file later using a text editor. You can then distribute the response file to multiple users with various login and password credentials. However, be aware that this can compromise your database security if you do not remove the password from the response file before distributing it.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 1: Encrypts the password in the response file, which ensures that your password is secure. This is the default behavior.</td>
<td></td>
</tr>
<tr>
<td>DSSUser=</td>
<td>Defines the MicroStrategy user name to log in to the project.</td>
<td></td>
</tr>
<tr>
<td>DSSPwd=</td>
<td>Defines the password for the MicroStrategy user name to log in to the project.</td>
<td></td>
</tr>
<tr>
<td>MDPrefix=</td>
<td>Defines a prefix for metadata repository tables used by the server definition.</td>
<td></td>
</tr>
<tr>
<td>TempTable=</td>
<td>Defines a prefix to be used for temporary tables used by the server definition.</td>
<td></td>
</tr>
<tr>
<td>UseAsDefault=</td>
<td>Defines whether the Intelligence Server definition is set as the default server definition to use for Intelligence Server, as determined by the following values:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• True: Defines the Intelligence Server definition as the default server definition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• False: Does not define the Intelligence Server definition as the default server definition</td>
<td></td>
</tr>
<tr>
<td>Port=</td>
<td>Defines the port used by the Intelligence Server.</td>
<td></td>
</tr>
<tr>
<td>RegisterAsService=</td>
<td>This option is only available on Intelligence Servers running on UNIX or Linux operating systems.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Defines whether Intelligence Server is registered as a service. Registering Intelligence Server as a service is determined by the following values:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 1: Registers Intelligence Server as a service. Performing this task requires a UNIX or Linux login with root level access and privileges.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 0: Does not register Intelligence Server as a service.</td>
<td></td>
</tr>
<tr>
<td>StartServerAfter Config=</td>
<td>Defines whether Intelligence Server is started after applying the configuration, as determined by the following values:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 1: Intelligence Server is started after successfully applying the configuration.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• 0: Intelligence Server is not started after applying the configuration.</td>
<td></td>
</tr>
</tbody>
</table>
Creating and configuring project sources

The response file parameters within the [Client] section create and configure project sources. The table below lists the available parameters and the functionality of available options for each parameter.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Client]</td>
<td>In this section you can configure the project source name. You can have more than one [Client] section. Additional client sections can be included as [Client1], [Client2], and so on.</td>
</tr>
<tr>
<td>Client=</td>
<td>Defines whether project sources are configured, as determined by the following values:</td>
</tr>
<tr>
<td></td>
<td>• 1: Configures project sources</td>
</tr>
<tr>
<td></td>
<td>• 0: Does not configure project sources</td>
</tr>
<tr>
<td>EncryptPassword=</td>
<td>Defines whether the password is encrypted in the response file, as determined by the following values:</td>
</tr>
<tr>
<td></td>
<td>• 0: The password is not encrypted in the response file, which enables you to modify the password in the response file later using a text editor. You can then distribute the response file to multiple users with various login and password credentials. However, be aware that this can compromise your database security if you do not remove the password from the response file before distributing it.</td>
</tr>
<tr>
<td></td>
<td>• 1: Encrypts the password in the response file, which ensures that your password is secure. This is the default behavior.</td>
</tr>
<tr>
<td>DataSource=</td>
<td>Defines the name of the new project source to create.</td>
</tr>
<tr>
<td>ConnType=</td>
<td>Defines the database connection type for a project source. The following connection types are supported:</td>
</tr>
<tr>
<td></td>
<td>• 2: Connects a project source to the metadata using an ODBC DSN (Windows only).</td>
</tr>
<tr>
<td></td>
<td>• 3: Connects a project source to the metadata through a MicroStrategy Intelligence Server (three-tier).</td>
</tr>
<tr>
<td>DSN=</td>
<td>If using connection type 2 (ConnType=2), defines the name of the ODBC database.</td>
</tr>
<tr>
<td>UserName=</td>
<td>If using connection type 2 (ConnType=2), defines the user name to connect to the ODBC database.</td>
</tr>
<tr>
<td>UserPwd=</td>
<td>If using connection type 2 (ConnType=2), defines the password to log in to the database.</td>
</tr>
<tr>
<td>ServerName=</td>
<td>If using connection type 3 (ConnType=3), defines the name of the MicroStrategy Intelligence Server to connect to.</td>
</tr>
<tr>
<td>Port=</td>
<td>If using connection type 3 (ConnType=3), defines the port number for the Intelligence Server when creating a server (three-tier) project source. The default port number for MicroStrategy Intelligence Server is 34952.</td>
</tr>
</tbody>
</table>
192 Connecting to a data source

For MicroStrategy users to be able to browse attribute elements and execute reports, a connection to a data warehouse must be created. A connection to other data sources can also support History Lists, statistics, and including data from multiple data sources into your MicroStrategy project.

Although metadata connectivity settings are stored in the server definition and registry of the MicroStrategy Intelligence Server machine, data source connectivity settings are stored in the metadata itself. After Intelligence Server connects to the metadata at startup, it loads all project, user, and connection information from the metadata. You can perform data source connection tasks from the Project Configuration Editor, which can be accessed by right-clicking a project and selecting Project Configuration.

The tasks described in this section require MicroStrategy Administrator privileges.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authentication=</td>
<td>The following authentication modes are supported:</td>
</tr>
<tr>
<td></td>
<td>• 1: Standard or login ID and password entered by the user</td>
</tr>
<tr>
<td></td>
<td>• 2: Network login ID (Windows authentication)</td>
</tr>
<tr>
<td></td>
<td>• 8: Guest account (Anonymous authentication)</td>
</tr>
<tr>
<td></td>
<td>• 16: LDAP authentication</td>
</tr>
<tr>
<td></td>
<td>• 32: Database login ID and password (database authentication)</td>
</tr>
<tr>
<td></td>
<td>• 128: Integrated authentication</td>
</tr>
<tr>
<td></td>
<td>For information on the available authentication modes, see the Authentication modes, page 181.</td>
</tr>
<tr>
<td>MDPrefix=</td>
<td>If using connection type 2 (ConnType=2), defines a prefix for metadata repository tables.</td>
</tr>
<tr>
<td>Timeout=</td>
<td>Defines and enforce a connection time out for inactive users connected to a project source. The following values are supported:</td>
</tr>
<tr>
<td></td>
<td>• 0: Defines that users are not disconnected from project sources due to inactivity.</td>
</tr>
<tr>
<td></td>
<td>• Numerical value greater than 0: A numerical value (in minutes) greater than 0 defines the amount of inactivity that is allowed before a user is automatically disconnected from a project source.</td>
</tr>
</tbody>
</table>

Connecting to a data source
Specifying warehouse connection information

A database instance is a MicroStrategy object, created in MicroStrategy Desktop by an administrator, that represents a connection to a data source. A database instance specifies connection information, such as the data source name, Login ID and password, and other data source specific information.

The steps to create the required components of a database instance are provided in the following sections: Creating a database instance, page 197, Creating a database connection, page 200, and Creating a database login, page 204.

When a project architect creates a project, the architect assigns a database instance to that project. A project specifies only one warehouse database instance at a time, but a database instance can be assigned to multiple projects. Since only one data source can be included in the project’s relational schema, all reports and documents return information from a single data source.

If you have a license for the new MultiSource Option feature, you can connect a project to multiple warehouse database instances. There can be multiple data sources that connect to the Warehouse Catalog for the project. Since these data source can be integrated as part of the project’s relational schema, all reports and documents can return information from multiple data sources. For information on accessing multiple data sources in a project, see the Project Design Guide.

Regardless of whether you have a license for the MultiSource Option, you can also extend a project’s access to multiple data sources through other MicroStrategy features. Freeform SQL, Query Builder, and supporting access through MicroStrategy to other MDX cube sources such as SAP BW, Hyperion Essbase, and Microsoft Analysis Services allows non-project database instances to be included and used in a project along with the warehouse database instances. For information on Freeform SQL and Query Builder, see the Advanced Reporting Guide. For information on MDX cube sources, see the MDX Cube Reporting Guide.

These non-project database instances can allow a project to connect to the data sources for the various features and additional data sources mentioned above, instead of accessing the data from the project’s relational schema. For more information on the Warehouse Catalog, see the MicroStrategy Project Design Guide.

The database instances that you create are separated into two categories:

- SQL data warehouses database instances, page 194
• **MDX cube database instances, page 195**

### SQL data warehouses database instances

A SQL data warehouse database instance is any database instance that connects to a database or other data source through SQL queries. More specifically, this covers database instances used for standard MicroStrategy reporting, Freeform SQL, Query Builder, data marts, and any other relational data source. You can also connect to History List and statistics tables through SQL data warehouse database instances. The SQL data warehouse database instances are available in the Project Configuration Editor, as shown below.

Selecting a database instance check box makes that database instance available in the project for standard MicroStrategy reporting, data marts, Query Builder, and Freeform SQL. If you have a license for the MultiSource Option, selecting a check box for a database instance also makes the database instance available from the Warehouse Catalog to be part of the project’s relational schema.

Database instances can be created as part of the Import Data feature. A database instance used for the Import Data feature is displayed with the ![icon](image.png) icon. These database instances are created with security permissions for the user that created them while using the Import Data feature. If you select one of these database instances to be
included as an available database instance in the project, it is recommended that you change the security permissions to a MicroStrategy user with administrative privileges. This includes taking ownership of the database instance and defining an appropriate access control list. This ensures that no changes are made to the database instance by other users, which could cause a loss of connectivity to the data source. For information on the Import Data feature, refer to the MicroStrategy Web online help.

The shading and color of a database instance in the list of relational database instances reflects how the database instance is being used in the project:

- **Blue text:** This is the warehouse database instance, as selected from the warehouse database instance drop-down list. There can only be one warehouse database instance for a project, because this database instance’s data is populated in the Warehouse Catalog to define the project’s relational schema. You cannot choose to disable the warehouse database instance for the project without first selecting a different warehouse database instance.

  If you have a license for the MultiSource Option, the primary database instance acts as the main source of data for a project and is used as the default database instance for tables added to the project.

  For information on the Warehouse Catalog as well as accessing multiple data sources with the MultiSource Option, see the *Project Design Guide*.

- **Bold text:** The project contains objects that are dependent on the database instance. You cannot choose to disable a database instance that has dependent objects for the project.

- **Normal text:** The database instance is not being used in the project.

Clearing the check box of a database instance removes the database instance from the project and deletes any unused Freeform SQL or Query Builder schemas. You can clear a database instance from a project only if there are no dependent objects in the project for the database instance. For more information on removing a database instance and related Freeform SQL and Query Builder schemas from a project, refer to the *MicroStrategy System Administration Guide*.

### MDX cube database instances

An MDX cube database instance is any database instance that connects to an MDX cube source, such as SAP BW, Hyperion Essbase, or Microsoft Analysis Services. For information on connecting to and reporting on these MDX cube
sources, refer to the *MDX Cube Reporting Guide*. The MDX cube database instances are available in the Project Configuration Editor, as shown below.

A database instance that has an MDX cube schema is represented with bold text. The shading and color of a database instance in the list of relational database instances reflects how the database instance is being used in the project:

- **Bold**: The project contains objects that are dependent on the database instance. You cannot choose to disable a database instance that has dependent objects for the project.
- **Normal**: The database instance is not being used in the project.

If you remove an MDX cube database instance from a project, you can delete any unused MDX cube schema objects. You can remove database instance from a project only if there are no dependent objects in the project for the database instance. For more information on removing a database instance and related MDX cube managed objects from a project, refer to the *System Administration Guide*.

For additional information on configuring MDX cube database instances, refer to the *MDX Cube Reporting Guide*. 
**MDX schema loading and maintenance**

You can click Schema Maintenance to perform various tasks for an MDX cube schema that is part of your project, as described below:

- You can choose when an MDX cube schema associated with a database instance is loaded for a project. By default, MDX cube schemas are loaded as needed when MDX cube reports are executed. You can also choose to load MDX cube schemas when Intelligence Server starts. For information on defining when MDX cube schemas should be loaded, refer to the *MDX Cube Reporting Guide*.

- When you integrate MDX cube sources into MicroStrategy, the data is integrated as an MDX cube schema. Once you integrate an MDX cube source into MicroStrategy, you can exchange the database instance used to connect to the MDX cube schema for a different database instance. This allows you to use different database instances with different login and connection information to access an MDX cube schema. For information on exchanging the database instance used to connect to the MDX cube schema, refer to the *MDX Cube Reporting Guide*.

**Creating a database instance**

Database instances are created and modified in the Database Instance Manager, which can be found by expanding Administration for a project.
source, then expanding **Configuration Managers**. When you choose to create a new database instance, the Database Instances Editor opens.

![Database Instances Editor](image)

You can also create a new database instance using the Database Instance Wizard that is available in the Database Instance Manager shortcut menu.

The Database Instances Editor has the following tabs:

- **General**—specifies the database instance name, connection type (data source platform or applicable data source), and default database connection.
  
  The database connection type you choose should match your data source and determines whether the database instance is a relational or an MDX cube database instance.

- **Advanced**—specifies the database name for intermediate table storage if a database other than the warehouse is used to store intermediate tables, as well as other options.
  
  The Advanced tab is not available for MDX cube database instances.

- **Job Prioritization**—specifies the job prioritization scheme for the instance and the number of prioritized connections.
To create a database instance

1. In MicroStrategy Desktop, log in to a project source with administrative privileges.

2. Expand Administration, then expand Configuration Managers, and then select Database Instances.

3. From the File menu, point to New, and then select Database Instance. The Database Instances Editor opens.

4. On the General tab, in the Database instance name field, type the name of the database instance.

5. In the Database connection type drop-down list, select the data source connection type according to the data source hosting your database.

   If you have upgraded from a previous version of MicroStrategy, you can click Upgrade to retrieve any database connection types that have been included since the previous version of MicroStrategy that you used.

6. On the Advanced tab, you can configure various options for the database instance, including:
   - Intermediate table storage: You can specify the database name and table name space to use when intermediate tables are created. Intermediate tables are created to support various queries.
   - Database gateway support: You can support backwards compatibility for database gateway support from MicroStrategy version 6.x.

      To enable database gateway support, select the Primary database instance check box, and then select a primary database instance from the drop-down list. The primary database instance is the database instance that should be used for element browsing against the selected table and for queries that do not require joins to other tables. For information on database gateway support, see the Project Design Guide.

   - Data mart optimization: You can support data mart optimization if the data source for the database instance is in the same data source that contains data marts.
To enable data mart optimization, select the **This database instance is located in the same warehouse as** check box, and then select a database instance from the drop-down list.

- **Table prefix**: If the tables in your data source use a table prefix, you can include the table prefix to identify the proper collection of tables. Click **Select** to select a table prefix or define a new table prefix.

- **ODBC Version**: You can define which ODBC version to use for the database instance, as described below:
  - **Use 2.0 ODBC Calls**: ODBC 2.0 was used in pre-9.0 versions of MicroStrategy. You can use this option for backward compatibility if your database management system does not support ODBC 3.x.
  - **Use 3.x ODBC Calls**: The support of ODBC 3.x is introduced in MicroStrategy 9.0. You should use this option if your database management system supports ODBC 3.x.

7 On the **Job Prioritization** tab, you can configure how jobs are prioritized for the database instance. For information on configuring job prioritization, see the **System Administration Guide**.

8 On the **General** tab, in the **Database connection (default)** pane, select the default data source connection and click **OK**.

If the necessary database connection does not exist, you can create one by clicking **New**. For steps to create a database connection, see **Creating a database connection** below.

### Creating a database connection

A **database connection** specifies the DSN and database login used to access the data source. A database instance designates one database connection as the default connection for MicroStrategy users; however, users and groups can be mapped to other database connections using connection mapping. For more details on connection mapping, see **User connection mapping, page 206**.

You create database connections in the Database Instances Editor by clicking **New** on the General tab. Any database connection created within the Database Instances Editor is available for use across all database instances in the project source. For more information on creating a database connection for MDX cube sources, refer to the **MDX Cube Reporting Guide**.
When you choose to create a new database connection, the Database Connections dialog box opens.

![Database Connections Dialog Box]

The Database Connections dialog box has different options depending on the database instance type.

- **SQL data warehouse database instances**
  - General—specifies the database connection name, the warehouse DSN, and the default database login.
  - Advanced—specifies the database driver mode, driver execution mode, and other miscellaneous warehouse connection settings.

- **MDX cube database instances**
  - General—specifies the database connection name, the default database login, and additional connection information that you must provide. For more information on creating a database connection for MDX cube sources, see the *MDX Cube Reporting Guide*.
  - Advanced—specifies the connection settings, additional connection string parameters, and connection caching settings.
The steps below show you how to create a database connection for a relational database instance. For information on creating a database connection for MDX cube sources, refer to the *MDX Cube Reporting Guide*.

**Prerequisites**

- A database instance has been created, as described in *Creating a database instance*, page 197.

---

**To create a database connection**

1. On the **General** tab, in the **Database connection name** box, type a name to identify the database connection.

2. In the **Local system ODBC data sources** pane, select the data source name for the data source.

3. On the **Advanced** tab, you can define various options per your requirements and the requirements of the database you are connecting to, including:

   - **Database driver mode**: Select one of the following database driver modes:
     
     - **Multi-process**: Each connection to the warehouse database is spawned as a separate process, identified in Windows Task Manager as `M8DBMPE.exe`. If one process fails, such as when a database access thread hangs or is lost, other processes are not affected.
     
     - **Multi-threaded**: All connections to the warehouse database are maintained inside the Intelligence Server process `MSTRSVR.exe`. All connections, SQL submissions, and data retrievals from the database are handled within this process.

MicroStrategy recommends setting all database drivers to multi-process mode. The robustness and stability which come with multi-process mode greatly overshadow any increased efficiency that may come with multi-threaded mode. Problems that appear random and sporadic in multi-threaded operation can often be resolved by switching to multi-process mode.
• **Driver execution mode:** Define the driver execution mode depending on the ODBC driver being used:
  
  – **Asynchronous Connection:** All statements allocated within the connection should be able to run SQL asynchronously.
  
  – **Asynchronous Statement:** For each statement, the asynchronous mode is explicitly set.
  
  – **Synchronous:** Only one statement executes at a time. This is the default value.
  
  Many newer ODBC drivers do not support asynchronous mode because the driver is capable of opening a new thread and executing a new query while simultaneously running an earlier query. The MicroStrategy Readme gives recommendations for the driver execution mode options that can be used for different ODBC drivers.
  
  • **Use extended fetch:** Select this check box to enable Intelligence Server to fetch blocks of data from the database into memory, instead of row-by-row.
  
  The MicroStrategy Readme recommends settings for ODBC drivers and whether to use the extended fetch feature.
  
  • **Use parameterized queries:** Select this check box to enable Intelligence Server to pass data to the database in blocks instead of row-by-row. For information on how parameterized queries can improve performance in MicroStrategy, see the *Project Design Guide*.
  
  • **Maximum cancel attempt time (sec):** Defines the maximum amount of time the MicroStrategy Query Engine waits for a successful attempt before it cancels a query. Values of 0 and -1 indicate no limit.
  
  • **Maximum query execution time (sec):** Defines the maximum amount of time a single pass of SQL can execute on the database. Values of 0 and -1 indicate no limit.
  
  • **Maximum connection attempt time (sec):** Defines the maximum amount of time Intelligence Server waits to connect to the database. Values of 0 and -1 indicate no limit.
  
  • **Additional connection string parameters:** Enables you to pass additional ODBC connection parameters to the database as part of the connection string. This is useful if you need to change ODBC defaults. Click **Preview** to see the entire connection string.
• **Table prefix**: Defines a table prefix that specifies the schema containing the tables to access.

• **Character set encoding for Windows drivers**: The options listed below are only relevant when Intelligence Server is running on a Windows machine:
  – **Non UTF-8**: Select this option if the ODBC driver returns information in a character encoding other than UTF-8.
  – **UTF-8**: Select this option if the ODBC driver returns information in UTF-8 character encoding. Drivers for Teradata databases may require UTF-8 encoding.

• **Character set encoding for UNIX drivers**: The options listed below are only relevant when Intelligence Server is running on a UNIX machine:
  – **Non UTF-8**: Select this option if the ODBC driver returns information in a character encoding other than UTF-8.
  – **UTF-8**: Select this option if the ODBC driver returns information in UTF-8 character encoding. Drivers for Teradata databases may require UTF-8 encoding.

• **Database connection caching settings**: Select one of the following database connection caching settings:
  – **Connection lifetime (sec)**: Defines the amount of time an active database connection remains open and cached on Intelligence Server. Values of 0 and -1 indicate no limit.
  – **Connection idle timeout (sec)**: Defines the amount of time an inactive connection to the database remains cached until it is terminated. Values of 0 and -1 indicate no limit.

4. On the **General** tab, in the **Default database login name** pane, select the default database login and click **OK**.

   If the necessary database login does not exist, you can create one by clicking **New**. For steps to create a database connection, see *Creating a database login* below.

---

### Creating a database login

A **database login** specifies the user ID and password used to access the data source. The database login overwrites any login information stored in the
DSN. A database connection designates one database login as the default login for MicroStrategy users, however users and groups can be mapped to other database logins using connection mapping.

Connection mapping is explained in User connection mapping, page 206.

You create database logins in the Database Connections dialog box by clicking New on the General tab. Any database login created within the Database Connections dialog box is available for use across all database connections in the project source.

MicroStrategy reporting and analysis features require a general set of database login permissions that can connect to and modify the data source and metadata, as described below:

• For the metadata, the Select, Insert, and Update permissions are required. Intermediate tables are created in the metadata for recursive search queries, which requires Create and Drop permissions as well. Updating the schema requires the Delete permission.

• For the data warehouse, the Select, Create, Insert, and Drop permissions are required.

When you choose to create a new database login, the Database logins dialog box opens.

Prerequisites

• A database instance has been created, as described in Creating a database instance, page 197.

• A database connection has been created, as described in Creating a database connection, page 200.
To create a database login

1. In the **Database Login** box, type the name of the database login.
2. In the **Login ID** box, type the user ID needed to access the data source.
3. In the **Password** box, type the password that is paired with the user ID you specified. Click **OK**.

Database logins are passed to the data source any time a user executes a report or browses attribute elements. Therefore, all database logins created in MicroStrategy Desktop must be also be created as valid logins in the data source.

User connection mapping

User connection mapping is the process of mapping MicroStrategy users to database connections and database logins. For MicroStrategy users to execute reports, they must be mapped to a database connection and database login.

MicroStrategy users link to database connections and logins using:

- The default database connection (and, therefore, default database login)
- Specialized maps to a database connection and/or database login (different than the default connection and login) for either a user or user group

You can map users to connections and logins in the Project Configuration Editor or Command Manager. For information about how connection maps are used, see the *System Administration Guide*.

MicroStrategy reporting and analysis features require a general set of database login permissions to connect to and modify the data warehouse and metadata, as described below:

- For the metadata, the Select, Insert, and Update permissions are required. Intermediate tables are created in the metadata for recursive search queries, which requires Create and Drop permissions as well. Updating the schema requires the Delete permission.
- For the data warehouse, the Select, Create, Insert, and Drop permissions are required.
Prerequisites

- A database instance has been created, as described in *Creating a database instance, page 197*.
- A database connection has been created, as described in *Creating a database connection, page 200*.
- A database login has been created, as described in *Creating a database login, page 204*.

To create a connection map

1. In Desktop, log in to a project.
2. Right-click the project and select **Project Configuration**. The Project Configuration Editor opens.
3. In the **Categories** list, expand the **Database Instances** category, and then select **Connection mapping**.
4. Right-click in the **Database instances - Connection mapping** pane, and select **New**. A new connection mapping is added.
5. You can define the connection mapping by specifying the information described below:
   - **Database Instance**: The database instance which connects to the data source required for the connection mapping.
   - **User**: The user or user group to apply the connection mapping to.
   - **Language**: The language of the data accessed by the connection mapping. You can use connection mappings to support data internationalization. For information on supporting data internationalization with connection mappings, see the *Project Design Guide*.
   - **Database connection**: The data source to connect to.
   - **Database Login**: The database login for the connection mapping.
6. Click **OK**.
Creating a project

Now you have configured Intelligence Server and are ready to create a project. There are various ways to create a project to get your MicroStrategy project started. The different methods to create a project are described in the Project Design Guide.

The MicroStrategy platform provides a Tutorial project, which is a sample data warehouse and demonstration project you can use to learn about the various features that MicroStrategy offers. It is ready to be used and requires no additional configuration tasks. To use the MicroStrategy Tutorial, refer to the Basic Reporting Guide for more information. To create a new project using your own data, see the Project Design Guide.

Configuring your MicroStrategy installation

To help guide the rest of your installation and configuration steps, refer to the section Installing and configuring MicroStrategy on Windows, page 87 in Chapter 1, Planning Your Installation, for an installation and configuration checklist.
Introduction

This chapter describes the procedure to deploy a project to your user community using MicroStrategy Web and Web Universal. The process of deploying the ASP.NET version of MicroStrategy Web on Windows with Microsoft Internet Information Services (IIS) is explained in detail.

Steps to deploy MicroStrategy Web Universal in a UNIX/Linux and Windows environment with various Web and application servers are also explained in detail. MicroStrategy Web Universal is platform-independent and can be deployed using different combinations of operating systems, Web servers, and application servers.

Web application servers are not MicroStrategy products, so detailed steps cannot be provided for every combination of application server and operating system. This chapter supplies instructions for a few of the most common combinations. The procedures for different operating systems are similar, but you should refer to the vendor-provided information and documentation for details specific to your configuration, or contact MicroStrategy Technical Support.

MicroStrategy Web simplifies the job of deploying to large user groups because end users’ machines only need a supported browser. MicroStrategy
Web can be accessed from any supported browser because no code must be downloaded. Working as a thin client, MicroStrategy Web provides the functionality that end users and power users require to take full advantage of the MicroStrategy product suite.

This chapter has the following sections:

- **Deploying with IIS (Windows), page 210**
  
  Deploying with IIS is the only setup given for MicroStrategy Web, which is an ASP.NET version for Windows. The other deployment procedures use MicroStrategy Web Universal, which is platform-independent and can be deployed with different Web and application servers.

- General steps to deploy MicroStrategy Web Universal, page 214
- Deploying with WebLogic and Apache (Solaris), page 216
- Deploying with WebSphere and IBM HTTP Server (AIX), page 234
- Deploying with Sun Java System (Solaris), page 246
- Deploying with Tomcat (Windows), page 257
- Deploying with Tomcat (Linux), page 264
- Deploying with SAP NetWeaver (Windows), page 270
- Deploying with Oracle 10g (Windows), page 275
- Deploying with JBoss (Windows), page 280
- Administering your MicroStrategy Web deployment, page 286
- Configuring your MicroStrategy installation, page 288

## Deploying with IIS (Windows)

Microsoft IIS can be used to deploy MicroStrategy Web or MicroStrategy Mobile Server:

- **Deploying MicroStrategy Web, page 211**
- **Deploying Mobile Server, page 212**
Deploying MicroStrategy Web

The ASP.NET version of MicroStrategy Web can be deployed with IIS only on Windows.

Prerequisites

- For information on supporting IIS 6 and IIS 7, see Supporting MicroStrategy products with IIS 6 on 64-bit Windows operating systems, page 70 and Supporting IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server, page 68.

- You must have administrative privileges to deploy MicroStrategy Web for your project. If this is the first time you are logging in and you have not changed the default MicroStrategy administrative login, you can use Administrator as the login with no password. After the first time, the user name and password should be changed for security purposes.

- The Microsoft Windows’ Users group must have read and execute permissions to all of the files within the MicroStrategy common files folder. This ensures that Internet Information Services has the required permissions to host MicroStrategy Web. By default, this folder is stored in the following directory location:

  - **32-bit Windows environments**: C:\Program Files\Common Files\MicroStrategy
  - **64-bit Windows environments**: C:\Program Files (x86)\Common Files\MicroStrategy

To connect MicroStrategy Web to your Intelligence Server

1. On the Windows Start menu, point to Programs, then to MicroStrategy, then to Web, and then choose Web Administrator. The MicroStrategy Web Administrator page opens. This is the page where you connect MicroStrategy Web to the Intelligence Server.

2. Type the name of your Intelligence Server in the Add a server manually box on the MicroStrategy Web Administrator page.

3. Click Connect. All projects loaded on the Intelligence Server are now available from MicroStrategy Web. Click the Home icon to see the list of projects loaded on the Intelligence Server you specified.
Send your users the URL:

http://webservername/MicroStrategy/asp/

where webservername is the name of the computer hosting your Web server. For example, if the name of your Web server machine is Web_Srv1, then the URL your users would use to access MicroStrategy Web would be

http://Web_Srv1/MicroStrategy/asp

You have manually connected MicroStrategy Web to the Intelligence Server.

You can also connect automatically whenever MicroStrategy Web Server or Intelligence Server starts.

---

To make MicroStrategy Web connect to the Intelligence Server automatically


2. Select the Automatically connect to Intelligence Server when Web Server or Intelligence Server is restarted option and click Save.

---

### Deploying Mobile Server

The ASP.NET version of MicroStrategy Mobile Server can only be deployed with IIS only on Windows.

**Prerequisites**

- For information on supporting IIS 6 and IIS 7, see Supporting MicroStrategy products with IIS 6 on 64-bit Windows operating systems, page 70 and Supporting IIS 7 or IIS 7.5 as a web server for MicroStrategy Web or Mobile Server, page 68.

- You must have administrative privileges to deploy MicroStrategy Mobile Server for your project. If this is the first time you are logging in and you have not changed the default MicroStrategy administrative login, you can use Administrator as the login with no password. After the first time, the user name and password should be changed for security purposes.

- The Users group for Microsoft Windows must have read and execute permissions to all of the files within the MicroStrategy common files
folder. This ensures that IIS has the required permissions to host MicroStrategy Mobile Server. By default, this folder is stored in the following directory location:

- **32-bit Windows environments**: C:\Program Files\Common Files\MicroStrategy
- **64-bit Windows environments**: C:\Program Files (x86)\Common Files\MicroStrategy

To connect MicroStrategy Mobile Server to your Intelligence Server

1. On the Windows Start menu, point to Programs, then to MicroStrategy, then to Mobile, then to Mobile Server, and then select Mobile Administrator. The MicroStrategy Mobile Server Administrator page opens. This is the page where you connect MicroStrategy Mobile Server to the Intelligence Server.

2. Type the name of your Intelligence Server in the Add a server manually box on the MicroStrategy Mobile Server Administrator page.

3. Click Connect.

4. Click Mobile Configuration to configure your MicroStrategy Mobile applications to communicate with Mobile Server and Intelligence Server. For steps on how to define this configuration, see the MicroStrategy Mobile Design and Administration Guide.

You have manually connected MicroStrategy Mobile Server to the Intelligence Server.

You can also connect automatically whenever MicroStrategy Mobile Server or Intelligence Server starts.

To make MicroStrategy Mobile Server connect to the Intelligence Server automatically


2. Select the Automatically connect to Intelligence Server when Mobile Server or Intelligence Server is restarted option.
3 Click **Save**.

**General steps to deploy MicroStrategy Web Universal**

After you have installed MicroStrategy Web Universal, you can deploy and configure it for your specific environment. The configuration and deployment steps are provided in the Web server and application server sections in this chapter. The table below lists the general steps for all environments.

<table>
<thead>
<tr>
<th>High-Level Deployment Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Log on to the application server by using the proper user name and password.</td>
</tr>
<tr>
<td>3 To increase the performance of the application before proceeding with the deployment, see the <em>Performance-based setup information</em> section, if available, for your environment and configure as necessary. Also, after deploying MicroStrategy Web Universal on your machine, there may be a few performance-based setup steps that you should complete.</td>
</tr>
<tr>
<td>4 Choose the desired deployment method.</td>
</tr>
<tr>
<td>5 Follow the deployment procedure.</td>
</tr>
<tr>
<td>6 Log on to the MicroStrategy Web Administrator Page.</td>
</tr>
<tr>
<td>7 Launch MicroStrategy Web Universal.</td>
</tr>
<tr>
<td>8 Start working with the application.</td>
</tr>
</tbody>
</table>

You must perform extra configuration steps to allow graphs to support non-Western European fonts on MicroStrategy Web Universal for a UNIX system. For more information, see *Graph and document support of non-Western European fonts*, page 530 of Appendix B, Troubleshooting.

The same general steps used to deploy MicroStrategy Web Universal can also be used to deploy MicroStrategy Mobile Server J2EE. Therefore, you can use the deployment steps for the various environments documented in this
chapter to deploy MicroStrategy Mobile Server J2EE, and substitute a few configuration requirements specific to deploying Mobile Server J2EE.

<table>
<thead>
<tr>
<th>High-Level Deployment Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Log on to the application server by using the proper user name and password.</td>
</tr>
<tr>
<td>3 To increase the performance of the application before proceeding with the deployment, see the Performance-based setup information section, if available, for your environment and configure as necessary. Also, after deploying MicroStrategy Mobile Server J2EE on your machine, there may be a few performance-based setup steps that you should complete.</td>
</tr>
<tr>
<td>4 Choose the desired deployment method.</td>
</tr>
<tr>
<td>5 Follow the deployment procedure.</td>
</tr>
<tr>
<td>6 Log on to the MicroStrategy Mobile Server Administrator Page.</td>
</tr>
<tr>
<td>7 From the MicroStrategy Mobile Server Administrator Page, configure your MicroStrategy Mobile applications to communicate with Mobile Server and Intelligence Server. For steps on how to define this configuration, as well as steps to deploy and configure a certificate server for Mobile Server, see the MicroStrategy Mobile Design and Administration Guide.</td>
</tr>
<tr>
<td>8 Start working with the application.</td>
</tr>
</tbody>
</table>

### Directory structure after deploying the WAR file

The following table shows the default directory structure after deploying the Web Universal WAR file in your application server.

<table>
<thead>
<tr>
<th>Directory</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>\help</td>
<td>Help and descriptor files</td>
</tr>
<tr>
<td>\images</td>
<td>All image files</td>
</tr>
<tr>
<td>\javascript</td>
<td>Interface JavaScript files</td>
</tr>
<tr>
<td>\jsp</td>
<td>Interface JSP code files</td>
</tr>
<tr>
<td>\style</td>
<td>Interface style files</td>
</tr>
<tr>
<td>WEB-INF</td>
<td>Configuration information for MicroStrategy Web</td>
</tr>
<tr>
<td>WEB-INF\classes</td>
<td>All resource files</td>
</tr>
<tr>
<td>WEB-INF\lib</td>
<td>All library files</td>
</tr>
<tr>
<td>WEB-INF\log</td>
<td>All log files, except for ncsportal.log files, which are created in the application server root folder</td>
</tr>
<tr>
<td>WEB-INF\tlds</td>
<td>All Tag Library Descriptor files</td>
</tr>
<tr>
<td>WEB-INF\xml</td>
<td>All MicroStrategy Web xml and xsl files</td>
</tr>
</tbody>
</table>
Deploying with WebLogic and Apache (Solaris)

This section provides information used to deploy and configure MicroStrategy Web Universal on the Oracle Solaris 9 operating system, using Apache as the Web server and Oracle WebLogic Server as the application server. It provides information for WebLogic 9.x and 10.3. You can also use the steps provided below to deploy MicroStrategy Mobile Server J2EE.

MicroStrategy certifies deploying MicroStrategy Web Universal with WebLogic 9.x only as an exploded directory.

This section includes the following information:

- **WebLogic paths and folder locations**: Default folder structure for each version of WebLogic.
- **Preconfiguration information**: Configuration that must occur before you begin deploying MicroStrategy Web Universal.
- **Deploying MicroStrategy Web Universal**: Instructions for deploying the application.
- **Re-deploy the application**: Instructions for re-deploying the application.
- **Performance-based setup information**: Optional configuration settings to increase the application’s performance.

The additional configuration steps are not required for MicroStrategy Web Universal to run, but these settings can increase its performance. Review the performance-based setup information prior to deploying the system to see if these changes are of interest to you.

**WebLogic paths and folder locations**

This section presents the default folder structure for each version of WebLogic, and provides the variable used throughout the rest of this chapter to represent the WebLogic mydomain folder path.

Each version of WebLogic is installed with a different default path to the WebLogic mydomain folder. When deploying MicroStrategy Web Universal, you must make some changes within the WebLogic folders. Thus, it is important to understand the WebLogic folder structure for the version of
WebLogic you are using. The following path reflects the default folder structure for WebLogic 9.x and 10.3:
WEBLOGIC_HOME/user_projects/domains/mydomain/

Note the following:

- WEBLOGIC_HOME is the WebLogic Server home path.
- The folder structures are configurable and your organization may have changed the default names or path.

Throughout this chapter, the WebLogic mydomain folder is referred to as WEBLOGIC_MYDOMAIN_FOLDER. This variable refers to the WebLogic mydomain folder in whatever location it resides on your system. The location of this variable is based on the version of WebLogic and whether your organization has changed the version’s default name or path.

Preconfiguration information

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Universal on your machine. This includes the following sections:

- Locating the WAR file
- Setting up Apache Web server to proxy requests to WebLogic
This section uses the configuration outlined in the following table. While your setup may vary slightly, for example, you may have different versions of these applications, the overall process remains the same.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Oracle Solaris 9 or 10 (on SPARC)</td>
</tr>
<tr>
<td>Web server</td>
<td>Apache 2.x</td>
</tr>
<tr>
<td>Application server</td>
<td>WebLogic 9.x or 10.3</td>
</tr>
<tr>
<td>JDK</td>
<td>Sun JDK 1.5.0</td>
</tr>
<tr>
<td></td>
<td>You can download this JDK from the following location:</td>
</tr>
<tr>
<td></td>
<td><a href="http://java.sun.com/j2se/1.5.0/download.html">http://java.sun.com/j2se/1.5.0/download.html</a></td>
</tr>
</tbody>
</table>

Note the following:

- For information on the version numbers supported or certified by MicroStrategy, see the MicroStrategy readme file.

- For information on installing these products, see http://www.oracle.com/technology/products/weblogic/integration/index.html

- Before you start the deployment process, locate the machine name and IP address.

**Locating the WAR file**

The MicroStrategy Web Universal application is packaged within a single file, called a WAR (Web ARchive) file, following J2EE specifications. You must deploy the WAR file to run the application in your application server environment.

The WAR file is placed in the folder you specified when installing MicroStrategy Web Universal. The default location when installing on 32-bit Windows environments is C:\Program Files\MicroStrategy\WebJSP. The default location when installing on 64-bit Windows environments is C:\Program Files (x86)\MicroStrategy\WebJSP. The default location when installing on UNIX or Linux is described in *Directory structure, page 127.*
Copy the WAR file to WEBLOGIC_MYDOMAIN_FOLDER. See WebLogic paths and folder locations, page 216 for information on the default folder structure.

To deploy MicroStrategy Web, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You can also refer to the section Deploying MicroStrategy Web Universal, page 220 below for more details on deploying the WAR file.

After deploying MicroStrategy Web, you can view the WEB-INF folder, which contains a subfolder named log. The log folder retains all the log files. For more information on the directory structure after deploying MicroStrategy Web, see Directory structure after deploying the WAR file, page 215.

Locating the Mobile Server WAR file

If you are deploying MicroStrategy Mobile Server J2EE, you must use the WAR file for Mobile Server J2EE. The WAR file is placed in the folder you specified when installing Mobile Server J2EE. You can then use the same processes used to deploy the MicroStrategy Web Universal WAR file to deploy the Mobile Server J2EE WAR file.

Setting up Apache Web server to proxy requests to WebLogic

You can have the Apache Web server and WebLogic Server running independently on the same machine, but to configure Apache to proxy the desired requests to the WebLogic Server, you must install a plug-in provided by WebLogic. Complete the instructions at the following URLs to install and configure the plug-in.

For WebLogic 10.3, the URL is:

http://e-docs.bea.com/wls/docs100/plugins/apache.html

Install the plug-in with the WebLogic installation in the following location:

WEBLOGIC_HOME/wlserver_10.3/server/plugin/solaris/sparc/
where `WEBLOGIC_HOME` is the path to the WebLogic Server.

To increase the performance of MicroStrategy Web Universal, you can complete additional setup configurations before the deployment. For more information, see *Performance-based setup information, page 229.*

**Deploying MicroStrategy Web Universal**

When your machine has been configured with the necessary settings, you can deploy MicroStrategy Web Universal with Apache and WebLogic. This involves the following steps:

1. **Deploying automatically (development mode).**

   - or -

2. **Deploying manually (production mode).**

3. **Accessing the MicroStrategy Web Administrator page.**

4. **Launching the project.**

The *Performance-based setup information* section provides information on additional settings to increase application performance. These additional settings are not required but can increase the performance of MicroStrategy Web Universal. Review this information prior to deployment to see if these options are of interest to you.

You can deploy MicroStrategy Web Universal using one of the following deployment methods:

- The automatic deployment feature is the easiest and fastest way. See *Deploying automatically (development mode), page 221.* Choose the development mode in the `/WEBLOGIC_MYDOMAIN_FOLDER/bin/startWebLogic.sh` file, within the WebLogic Server folder structure.

- The manual deployment feature can be used for environments where the server is running in production mode and the automatic deployment is turned OFF. For more information, see *Deploying manually (production mode), page 224.*
Deploying automatically (development mode)

When automatic deployment is set to ON, as soon as you place a WAR file in the `/WEBLOGIC_MYDOMAIN_FOLDER/autodeploy` folder, the application is automatically deployed.

With this method you can deploy from:

- A duplicate WAR file. When you deploy from a duplicate WAR file, you are required to manually configure the `web.xml` file within the WAR file to allow access to certain folders. Once this configuration is complete and the WAR file is recompiled, MicroStrategy Web Universal can be deployed using the single WAR file.

- An exploded directory where all the files contained in the WAR file were extracted. When you deploy from an exploded directory, all of the files and folders within the WAR file are exposed to WebLogic. This allows WebLogic access to the required folders so that it can make any necessary configuration changes to files in the exploded directory.

---

To automatically deploy MicroStrategy Web Universal from a duplicate WAR file

1. Locate the `MicroStrategy.war` file in the MicroStrategy Web Universal Deployment Directory you specified during installation. For more information, see Installation procedures on UNIX and Linux, page 112.

   If you are deploying MicroStrategy Mobile Server J2EE instead, locate the `MicroStrategyMobile.war` file. Use this WAR file for the rest of the steps in this procedure.

2. Rename the WAR file to a name you can easily identify and remember. This name is the `context_name` used in the uniform resource locator (URL) to access the file. This step is optional.

   If you do not change the name of the file, remember to replace `context_name` with MicroStrategy when accessing the application from the URL.
To modify the web.xml file for multiple MicroStrategy Web Universal deployments

3 If you are deploying more than one MicroStrategy Web Universal environment on the same WebLogic application server, prior to deployment, you must modify the web.xml file as described below:

a Unzip the WAR file by using the following command:

```
#jar -xvf MicroStrategy.war
```

b Open the web.xml file located in the /WEB-INF directory.

c Modify the contextPath parameter. By default, this parameter does not have a value. Type a unique string for the value of the contextPath parameter. For example, type WebDep2.

d Save the web.xml file.

e Zip the WAR file by using the following command:

```
#jar -cvf MicroStrategy.war *
```

To deploy the WAR file

4 Transfer the WAR file to the following directory:

```
/WEBLOGIC_MYDOMAIN_FOLDER/autodeploy
```

The application is automatically deployed. To add and connect to an Intelligence Server, see Configuring the MicroStrategy Web Administrator page, page 226.

To increase the performance of MicroStrategy Web Universal, you can configure additional settings after deployment. For more information, see Performance-based setup information, page 229.

To automatically deploy MicroStrategy Web Universal from an exploded directory

1 The WAR file must be uncompressed by the same Solaris user who started the application.

Locate the MicroStrategy.war file. It is located in the MicroStrategy Web Universal Deployment Directory you specified during installation. For more information, see Installation procedures on UNIX and Linux, page 112.
If you are deploying MicroStrategy Mobile Server J2EE instead, locate the MicroStrategyMobile.war file. Use this WAR file for the rest of the steps in this procedure.

2 Create the following new folder:

/home/username/context_folder

where username is your account name used to access the Web server machine, and context_folder is the name of the new folder.

You can create the new folder anywhere except in the following location:

/WEBLOGIC_MYDOMAIN_FOLDER/autodeploy

3 Copy the WAR file to the new folder.

4 To explode the WAR file inside the folder you created, run the following command:

# jar -xvf MicroStrategy.war

5 Delete the MicroStrategy.war file by using the following command:

# rm MicroStrategy.war

6 Move the folder to the autodeploy folder with the following commands:

# cd..

# mv context_folder
/WEBLOGIC_MYDOMAIN_FOLDER/autodeploy

The application is automatically deployed. To add and connect to an Intelligence Server, see Configuring the MicroStrategy Web Administrator page, page 226.

To increase the performance of MicroStrategy Web Universal, you can configure additional settings after deployment. For more information, see Performance-based setup information, page 229.
Deploying manually (production mode)

With manual deployment you can deploy MicroStrategy Web Universal from:

- A duplicate WAR file. When you deploy from a duplicate WAR file, you are required to manually configure the web.xml file within the WAR file to allow access to certain folders. Once this configuration is complete and the WAR file is recompiled, MicroStrategy Web Universal can be deployed using the single WAR file.

- An exploded directory where all the files contained in the WAR file were extracted. When you deploy from an exploded directory, all of the files and folders within the WAR file are exposed to WebLogic. This allows WebLogic to access the required folders to perform any necessary configurations to files in the exploded directory.

Perform the deployment in the /WEBLOGIC_MYDOMAIN_FOLDER/autodeploy directory.

To manually deploy MicroStrategy Web Universal from a duplicate WAR file

1. Locate the MicroStrategy.war file. It is located in the MicroStrategy Web Universal Deployment path, as described in Directory structure, page 127.

   If you are deploying MicroStrategy Mobile Server J2EE instead, locate the MicroStrategyMobile.war file. Use this WAR file for the rest of the steps in this procedure.

To modify the web.xml file for multiple MicroStrategy Web Universal deployments

2. If you are deploying more than one MicroStrategy Web Universal environment on the same WebLogic application server, prior to deployment, you must modify the web.xml file as described below:

   a. Unzip the WAR file by using the following command:

      #jar -xvf MicroStrategy.war

   b. Open the web.xml file located in the /WEB-INF directory.

   c. Modify the contextPath parameter. By default, this parameter does not have a value. Type a unique string for the value of the contextPath parameter. For example, type WebDep2.
d  Save the web.xml file.

e  Zip the WAR file by using the following command:

```
#jar -cvf MicroStrategy.war *
```

To deploy the WAR file

3  Transfer the WAR file to the 
/WEBLOGIC_MYDOMAIN_FOLDER/autodeploy directory.

4  Open the WebLogic Server Administration Console (WLS Admin Console) by typing the following address:

```
http://IP address:port/console/
```

where IP address is the IP address of the machine on which you installed the WebLogic application server and port is the port number for the WebLogic application server.

5  Type a valid user ID and password at the prompt. The user ID and password are the ones you specified when installing the WebLogic Server on your machine.

6  To complete this operation, see Configuring from the WebLogic Server Administration Console, page 226.

To manually deploy MicroStrategy Web Universal from the exploded directory

1  Locate the MicroStrategy.war file. It is located in the MicroStrategy Web Universal Deployment Directory you specified during installation. For more information, see Installation procedures on UNIX and Linux, page 112.

   If you are deploying MicroStrategy Mobile Server J2EE instead, locate the MicroStrategyMobile.war file. Use this WAR file for the rest of the steps in this procedure.

2  Create a folder in the /WEBLOGIC_MYDOMAIN_FOLDER/autodeploy directory and transfer the WAR file to this directory.

3  Unzip the WAR file using the following command:

```
#jar -xvf MicroStrategy.war
```
4 Open the WebLogic Server Administration Console by accessing the following address:

http://IP address:port/console/

where IP address is the IP address of the machine on which you installed the WebLogic application server and port is the port number for the WebLogic application server.

5 Type a valid user ID and password at the prompt. The user ID and password are the ones you specified when installing the WebLogic Server on your machine.

6 To complete this operation, see Configuring from the WebLogic Server Administration Console below.

Configuring from the WebLogic Server Administration Console

To configure from the WebLogic Server Administration Console, refer to your WebLogic Server Administration Console documentation on steps to install a web application.

Once you have installed MicroStrategy Web Universal as a WebLogic Server Administration Console web application, you have completed the steps required to deploy MicroStrategy Web Universal.

To launch MicroStrategy Web Administrator and to configure MicroStrategy Intelligence Server, see Configuring the MicroStrategy Web Administrator page, page 226.

To increase the performance of MicroStrategy Web Universal, you can configure additional settings after deployment. For more information, see Performance-based setup information, page 229.

Configuring the MicroStrategy Web Administrator page

Before you start MicroStrategy Web Universal, you must configure the MicroStrategy Web Administrator page, which is the mstrWebAdmin servlet.
To access the MicroStrategy Web Administrator page

1 The address to launch MicroStrategy Web is different for each deployment method. The following table lists the URL to access MicroStrategy Web Administrator as well as MicroStrategy Mobile Server Administrator.

   The servlet names are case-sensitive. Make sure to use the correct case when typing the `mstrWebAdmin` name. If the application server is enabled with security, a dialog box related to the administrator authentication opens.

<table>
<thead>
<tr>
<th>Deployment Method</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automatic deployment</td>
<td>Access the Administrator page from a web browser using this URL:</td>
</tr>
<tr>
<td></td>
<td>• For Web Universal: <a href="http://IPaddress:7001/context_folder/">http://IPaddress:7001/context_folder/</a></td>
</tr>
<tr>
<td></td>
<td>servlet/mstrWebAdmin</td>
</tr>
<tr>
<td></td>
<td>• For Mobile Server J2EE: <a href="http://IPaddress:7001/context_folder/">http://IPaddress:7001/context_folder/</a></td>
</tr>
<tr>
<td></td>
<td>servlet/MobileServlet</td>
</tr>
<tr>
<td></td>
<td>where <code>context_folder</code> is the name of the folder where the application was exploded and</td>
</tr>
<tr>
<td></td>
<td><code>IPaddress</code> is the IP address of your machine.</td>
</tr>
<tr>
<td>Manual deployment</td>
<td>Access the Administrator page from a browser using this address:</td>
</tr>
<tr>
<td></td>
<td>• For Web Universal: <a href="http://IPaddress:7001/name/servlet/">http://IPaddress:7001/name/servlet/</a></td>
</tr>
<tr>
<td></td>
<td>mstrWebAdmin</td>
</tr>
<tr>
<td></td>
<td>• For Mobile Server J2EE: <a href="http://IPaddress:7001/name/servlet/">http://IPaddress:7001/name/servlet/</a></td>
</tr>
<tr>
<td></td>
<td>MobileServlet</td>
</tr>
<tr>
<td></td>
<td>where <code>IPaddress</code> is the IP address of your machine. Replace the <code>name</code> variable with the</td>
</tr>
<tr>
<td></td>
<td>name you specified in the deployed name field when configuring from WebLogic Server</td>
</tr>
<tr>
<td></td>
<td>Administration Console, for example, <code>MyWebApp</code>.</td>
</tr>
</tbody>
</table>

2 Type the same user ID and password that was used to start the WebLogic Server on your machine.

   In WebLogic, the deployment of MicroStrategy Web Universal automatically associates the WebLogic administrative user with the MicroStrategy Web Universal administrator. The WebLogic administrative user is the user who has permissions to start the WebLogic Server on a given machine.

3 After you are authenticated, the MicroStrategy Web Administrator page appears. Add and connect to an Intelligence Server.

   If you are deploying MicroStrategy Mobile Server J2EE, after you are authenticated, the MicroStrategy Mobile Server Administrator page
appears. Add and connect to an Intelligence Server. Once connected, click **Mobile Configuration** to configure your MicroStrategy Mobile applications to communicate with Mobile Server and Intelligence Server. For steps on how to define this configuration, see the *MicroStrategy Mobile Design and Administration Guide*. Creating a configuration completes the steps required to deploy Mobile Server J2EE.

4) Launch the MicroStrategy Web Universal project, as described below.

**Launching the project**

The address to launch MicroStrategy Web Universal is different for each deployment method. The table below lists the URL you can use to access MicroStrategy Web Universal.

<table>
<thead>
<tr>
<th>Deployment Method</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automatic deployment</td>
<td><strong><a href="http://IPaddress:7001/context_folder/servlet/mstrWeb">http://IPaddress:7001/context_folder/servlet/mstrWeb</a></strong> where <strong>context_folder</strong> is the name of the folder where the application was exploded and <strong>IPaddress</strong> is the IP address of your machine.</td>
</tr>
<tr>
<td>Manual deployment</td>
<td><strong><a href="http://IPaddress/name/servlet/mstrWeb">http://IPaddress/name/servlet/mstrWeb</a></strong> where <strong>IPaddress</strong> is the IP address of your machine. Replace the <strong>name</strong> variable with the name you specified in the deployed name field when configuring from WebLogic Server Administration Console, for example, MyWebApp.</td>
</tr>
</tbody>
</table>

**Re-deploy the application**

If you have already deployed MicroStrategy Web Universal with WebLogic and you change any parameters in the `web.xml` file, you must re-deploy the application using the WebLogic Server Administration Console. This allows the changes to take effect in the deployed application. To re-deploy MicroStrategy Web Universal, refer to your WebLogic Server Administration Console documentation on steps to re-deploy (update) a web application.
Performance-based setup information

The performance of MicroStrategy Web Universal can be increased by configuring it on various component levels. These additional setup settings are not required, but if you want to increase the performance of MicroStrategy Web Universal, some changes must be done before or after the deployment procedure. This section provides the following configurations:

- Setting the Java heap size
- Precompiling JSP files
- Disable/relax auto-reload parameters
- Configuring Apache Web server to serve static files

Setting the Java heap size

The Java heap size for the WebLogic Server can be increased by modifying the MEM_ARGS variable in the startWebLogic.sh script.

To increase the Java heap size

1. Open the startWebLogic.sh script from /WEBLOGIC_MYDOMAIN_FOLDER/bin/startWebLogic.sh.

2. Define the following line in the script:

   MEM_ARGS="-Xms512m -Xmx1024m"

   This line reflects an initial Java heap size of 512 MB. MicroStrategy recommends the initial java heap size be set at a minimum of 512MB, assuming the machine has enough memory space.

3. Stop and start the application server.

Precompiling JSP files

To avoid the time taken to load the Web pages in the application server when you access it for the first time, you must precompile the Java Server Pages (JSP) files before deploying the application. Do this by setting the application
server to load all the pages in the application before deployment. Thus, when you connect for the first time, the pages are already loaded and the performance is better.

---

**To precompile the JSP files**

1. Open the `weblogic.xml` file located in the `/WEB-INF` directory.

2. In the `jsp-descriptor` section, set the `keepgenerated` and the `precompile` parameters to TRUE, as follows:

   ```xml
   <jsp-descriptor>
     ...
     <jsp-param>
       <param-name>keepgenerated</param-name>
       <param-value>TRUE</param-value>
     </jsp-param>
     <jsp-param>
       <param-name>precompile</param-name>
       <param-value>TRUE</param-value>
     </jsp-param>
     ...
   </jsp-descriptor>
   
3. Save the file.

**Disable/relax auto-reload parameters**

To disable/relax auto-reload parameters, complete the following:

- *Set the pageCheckSeconds parameter*
- *Set the WebLogic Reload Period parameter*

Each parameter is explained below.
Set the pageCheckSeconds parameter

The `pageCheckSeconds` parameter sets the interval, in seconds, at which the WebLogic Server checks to see if JSP files have changed and need recompiling. Dependencies are also checked and recursively reloaded if changed.

You can set the following values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Pages are checked on every request.</td>
</tr>
<tr>
<td>-1</td>
<td>The page is not checked until the server is restarted. Any classes used by the JSP page that live in the servlet classpath are also reloaded.</td>
</tr>
<tr>
<td>n</td>
<td>Interval (in seconds) in which WebLogic Server checks if JSP files have changed. For example, if this is set to 1, WebLogic checks the pages every second to see if the JSP has changed and needs recompiling.</td>
</tr>
</tbody>
</table>

**To set the pageCheckSeconds parameter**

1. Open the `weblogic.xml` file located in the `/WEB-INF` directory.

2. In the `jsp-descriptor` section, set the `pageCheckSeconds` parameter value. For example, the following code sets the value to -1:

   ```xml
   <jsp-descriptor>
     ...
     <jsp-param>
       <param-name>pageCheckSeconds</param-name>
       <param-value>-1</param-value>
     </jsp-param>
     ...
   </jsp-descriptor>
   ```

3. Save the file.
Set the WebLogic Reload Period parameter

In WebLogic, the Reload Period parameter sets how often WebLogic checks whether a servlet has been modified. If the servlet has been modified, WebLogic reloads it. As the MicroStrategy Web Universal servlets do not change after they have been deployed, MicroStrategy recommends that you disable the reload period by setting it to -1. A value of -1 means never reload, and a value of 0 means always reload.

Use the appropriate procedure below, depending on whether you have MicroStrategy Web Universal deployed as a duplicate WAR file.

To set the WebLogic Reload Period

1. Open the `weblogic.xml` file located in the `/WEB-INF` directory.

2. In the `container-descriptor` section, set the `servlet-reload-check-secs` parameter value. For example, the following code sets the value to -1:

   ```xml
   <container-descriptor>
   <servlet-reload-check-secs>-1</servlet-reload-check-secs>
   </container-descriptor>
   
   3. Save the file.

Configuring Apache Web server to serve static files

Because Web servers are tuned to effectively serve static files, the perceived performance of MicroStrategy Web Universal is significantly enhanced if image, style sheet, and JavaScript files are served via the Apache Web server, and the WebLogic Server handles only the servlet requests. Do this by editing two main parameters, `Alias` and `MatchExpression`, in the Apache configuration file `httpd.conf`.

- The `Alias` parameter is used to create a virtual directory in the Apache Web server. The virtual directory is needed to serve static files such as images, style sheets, and JavaScript.

- The `MatchExpression` parameter is used to configure the Apache plug-in so that the WebLogic Server handles only the servlet requests.
To configure the Apache Web server to serve static files

1. To change the Alias parameter, add the following lines in the httpd.conf file:

```
Alias /MicroStrategy/images/"/WEBLOGIC_MYDOMAIN_FOLDER/autodeploy/MicroStrategy/images/"

<Directory "/WEBLOGIC_MYDOMAIN_FOLDER/autodeploy/MicroStrategy/images">
  Options Indexes MultiViews
  AllowOverride None
  Order allow, deny
  Allow from all
</Directory>
```

These code excerpts assume the application name is MicroStrategy. See Deploying with WebLogic and Apache (Solaris) for information on default folder structure.

2. Repeat the previous step for the JavaScript and style sheet folders, replacing the word images in the previous code with the folder name where the JavaScript and style sheet files are located.

3. Change the MatchExpression parameter by typing */servlet/* in the MatchExpression parameter. For example,

```
<IfModule mod_weblogic.c>
  WebLogicHost 10.15.133.56
  WebLogicPort 7001
  MatchExpression */servlet/*
</IfModule>
```

4. Stop and start the Apache Web server using the commands `apachectl start` and `apachectl stop`.

The Web server now serves image (GIF), style sheet (CSS), JavaScript, and all other static files, thus reducing the load on the application server and increasing the application’s performance.
Deploying with WebSphere and IBM HTTP Server (AIX)

This section provides information used to deploy and configure MicroStrategy Web Universal on an AIX machine using the WebSphere 6.1 Server and the IBM HTTP Web Server. You can also use the steps provided below to deploy MicroStrategy Mobile Server J2EE.

This section includes the following information:

- **Preconfiguration information**: Configuration that must occur before you begin deploying MicroStrategy Web Universal.

- **Deploying MicroStrategy Web Universal**: Instructions for deploying the application.

- **Performance-based setup information**: Optional settings to increase the application’s performance.

These additional settings are not required, but can increase the performance of MicroStrategy Web Universal. Review this information prior to deployment to see if any of these options are of interest to you.

### Preconfiguration information

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Universal on your machine. Preconfiguration includes the following step:

- **Locating the WAR file**

This section also provides additional setup information for the machine where the application server and Web server are already installed.
This section uses the following configuration. While your configuration may vary slightly, for example, you may have different versions of these applications, the overall process remains the same.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>AIX 5.3 or 6.1 (on IBM POWER Architecture)</td>
</tr>
<tr>
<td>Web server</td>
<td>IBM HTTP Server 6.0.x</td>
</tr>
<tr>
<td>Application server</td>
<td>IBM WebSphere application server 6.1 or 7.0</td>
</tr>
<tr>
<td>JDK</td>
<td>IBM JDK 1.5.0 or IBM JDK 1.6.0</td>
</tr>
<tr>
<td></td>
<td>You can download these JDKs from the following location: <a href="http://www.ibm.com/developerworks/java/jdk/aix/service.html">http://www.ibm.com/developerworks/java/jdk/aix/service.html</a></td>
</tr>
</tbody>
</table>

For information on the version numbers certified and supported by MicroStrategy, see the MicroStrategy readme file.

**Locating the WAR file**

The MicroStrategy Web Universal application is packaged within a single file, called a WAR (Web ARchive) file, following the J2EE specifications. You must deploy this file to run the application in your application server environment.

**To locate the WAR file**

1. Locate the `MicroStrategy.war` file in the MicroStrategy Web Universal Deployment Directory you specified during installation. For more information, see *Installation procedures on UNIX and Linux*, page 112.

   If you are deploying MicroStrategy Mobile Server J2EE instead, locate the `MicroStrategyMobile.war` file. Use this WAR file for the rest of the steps in this procedure.

2. Copy the WAR file to the `WAS_HOME/installableApps` directory, where `WAS_HOME` is the WebSphere 6.1 application server home path.
The deployment process in this chapter assumes that `MicroStrategy.war` file is stored in this directory.

To deploy the WAR file, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this chapter.

After deploying `MicroStrategy.war` file, you can view the WEB-INF folder, which contains a subfolder named `log`. The `log` folder retains all the log files, except for the `ncsportal.log` files. They are created in the Application server root folder. For more information on the directory structure after deploying the WAR file, see Directory structure after deploying the WAR file, page 215.

**Locating the Mobile Server WAR file**

If you are deploying MicroStrategy Mobile Server J2EE, you must use the WAR file for Mobile Server J2EE. The WAR file is placed in the folder you specified when installing Mobile Server J2EE. You can then use the same processes used to deploy the MicroStrategy Web Universal WAR file to deploy the Mobile Server J2EE WAR file.

**Deploying MicroStrategy Web Universal**

Once your machine has the necessary settings configured, you can deploy MicroStrategy Web Universal on the WebSphere-AIX machine. Deployment involves the following steps:

1. *Launching the WebSphere Administrative Console*
2. *Starting the WebSphere default application server*
3. *Installing the Enterprise Application*
4. *Regenerating plugin-cfg.xml*
5. *Restarting the application server*
6. *Accessing the MicroStrategy Web Administrator page*
7  Launching the project

The **Performance-based setup information** section provides information on additional settings to increase application performance. These additional settings are not required, but can increase the performance of MicroStrategy Web Universal. Review this information prior to deployment to see if any of these options are of interest to you.

**Launching the WebSphere Administrative Console**

The WebSphere Administrative Console can be accessed only if the WebSphere server is started on the machine.

---

**To start the Websphere Application Server**

1  Execute the `startServer` script as follows:

   ```
   cd WAS_HOME/bin
   # ./startServer.sh SERVER_NAME
   ```

   Typically, `server1` is the default server name in WebSphere 6.1.

2  Ensure that the Administrative Server has started successfully. Execute the following commands:

   ```
   cd WAS_HOME/bin
   # ./serverStatus.sh -all
   ```

---

**To launch the WebSphere Administrative Console**

In a Web browser, type the URL for the administrative console.

- The default port in WebSphere 6.1 is 9060 for HTTP.
  
  `http://HostName:9060/admin`
Starting the WebSphere default application server

After you launch the WebSphere Administrative Console, you can deploy MicroStrategy Web Universal by starting the default application server.

This is applicable for WebSphere Network Deployment Edition or WebSphere Enterprise Edition. For WebSphere Express or WebSphere Base Editions, there is no distinction between an administrative server and a default server. The StartServer.sh command starts the default application server automatically.

To start the default application server

When the WebSphere Administrative Console opens, a tree view is displayed.

1. Expand the Servers node, or click the link to expand the view.

2. Click the Applications Servers link. A table listing the application servers displays to the right of the navigation tree. This area is the Workspace.

3. Select the box next to the application server to start.

4. Click Start above the table.

Installing the Enterprise Application

To install the Enterprise Application

1. Expand Applications, and then Enterprise Applications to display a list of installed applications.

2. Click Install.

Preparing for the application installation

The following steps describe the settings that must be specified for the installation.
To specify settings for the installation

1 Specify the path to the WAR file by selecting the Server Path option and then type in the full path for the location of the MicroStrategy.war file. Type a suitable name for ContextRoot, which is case-sensitive. Do not include .war in the name for ContextRoot as this can cause errors when attempting to start the application.

The URL to access MicroStrategy Web Universal (http://machine-name/ContextRoot/servlet/mstrWeb) contains ContextRoot, which should be replaced by any name of your choice. For example, you can use the default name of the WAR file, which is MicroStrategy.

If you are deploying MicroStrategy Mobile Server J2EE instead, type the full path for the MicroStrategyMobile.war file.

2 Click Next.

3 Select Do not override existing bindings.

4 For the Default virtual host name for Web modules, select default host.

5 Click Next. The Provide options to perform the installation page opens. In the screens that follow, you are selecting settings that are used during the installation.

6 Click Next.

7 Make the following changes:
   
   - Select Enable pre-compile of JSPs.
   - Specify the value for the Directory to Install Application as
     \${APP_INSTALL_ROOT}/DefaultNode
   - Specify an Application Name of your choice.
   - Type the value of -1 as the Reload Interval.

8 Click Next. The Map virtual hosts for web modules page opens.

9 Select Web Tier (default host) and click Next. The Map modules to application servers page opens.

10 Select the application server and click Apply.
11 Select **Web Tier** (selected application server) and click **Next**.

12 Map the **admin** role to the users or groups that will be given the MicroStrategy Web Administrator privileges. Role mapping enables the mapped users to access the MicroStrategy Web Administrator page.

   Security must be enabled for the WebSphere Server for this feature to work.

13 Click **Next**. The Summary Page opens.

14 Review the summary and click **Finish**. A message appears stating that the installation and precompilation of JSPs was successful. Save the changes to the master repository.

### Regenerating plugin-cfg.xml

To regenerate plugin-cfg.xml

1 Select **Environment**, and then click **Update WebServer Plugin**.

2 Click **OK**, and then click **Save to master configuration**.

### Restarting the application server

This section explains how to stop and start the application server. Performing these steps stops and starts all the applications running on the application server. To stop and start only the application in which you are working, see *To start the Web module, page 241.*

To restart the application server

1 Expand the Servers node.
2  Click the **Application Servers** link. A table listing the application servers and an icon indicating their status appears:

- red: stopped
- green: started

3  Select the box next to the application server you want to stop, and click **Stop**. The status icon changes from green to red.

4  Select the application server you want to start and click **Start**. The application server starts and the status icon changes to green.

To stop and start the application server in Websphere Express and Websphere Base editions, use the following commands:

- `stopServer.sh server1` to stop the application server
- `startServer.sh server1` to start the application server.

### Starting the Web module

This process starts only the Web application, rather than all the applications running on the application server. To stop and start all applications, see *Restarting the application server*, above.

**To start the Web module**

1  Expand **Applications**.

2  Click the **Enterprise Applications** link. A list of installed applications are displayed in the table to the right, along with icons indicating their status:

- red: stopped
- green: started

3  Select the box next to the application to start and click **Start**.

### Accessing the MicroStrategy Web Administrator page

The MicroStrategy Web Administrator page (mstrWebAdmin servlet) is accessible only to users with an **admin** role. To create the set of users and
passwords that are authorized to access the administrator page, you must create the necessary role mapping between these users and the admin role for the MicroStrategy Web Universal application. The steps to perform this setup are given above in the section *Preparing for the application installation, page 238*. For more information, you can refer to your IBM documentation.

Before you start MicroStrategy Web Universal, you must configure the MicroStrategy Web Administrator page, which is the mstrWebAdmin servlet. If you are deploying Mobile Server J2EE, you can use the steps below to configure the MicroStrategy Mobile Server Administrator page, which is the MobileServlet servlet.

---

**To access the MicroStrategy Web Administrator page**

1. Access the servlet by typing the following URL in a Web browser:
   - For Web Universal: http://IPAddress/ContextRoot/servlet/mstrWebAdmin
   - For Mobile Server J2EE: http://IPAddress/ContextRoot/servlet/MobileServlet

   where ContextRoot is the name you provided for the ContextRoot for Web Module box in the section *Preparing for the application installation*. For example, the default name of the WAR file, which is MicroStrategy.

   The servlet names are case-sensitive. Use the correct case when typing the mstrWebAdmin or MobileServlet name. If the application server is enabled with security, a dialog box related to the administrator authentication opens.

2. Type the user ID and password assigned with the admin role.

3. After you are authenticated, the MicroStrategy Web Administrator page opens. Use it to add and connect to an Intelligence Server.

   If you are deploying MicroStrategy Mobile Server J2EE, after you are authenticated, the MicroStrategy Mobile Server Administrator page appears. Add and connect to an Intelligence Server. Once connected, click **Mobile Configuration** to configure your MicroStrategy Mobile applications to communicate with Mobile Server and Intelligence Server. For steps on how to define this configuration, see the *MicroStrategy Mobile Design and Administration Guide*. Creating a configuration completes the steps required to deploy Mobile Server J2EE.
Proceed to launch the MicroStrategy Web Universal project. For more information, see *Launching the project* immediately below.

**Launching the project**

After configuring the MicroStrategy Web Administrator page, you must follow the steps described in this section to launch your project.

---

**To launch the project**

1. Start the Apache Web server by using the following command:

   ```
   # /usr/HTTPServer/bin/apachectl start
   ```

   For example, if the IBM HTTP server is installed in the default location `/usr/IBMIHS`, then use the following command:

   ```
   /usr/IBMIHS/bin/apachectl
   ```

2. In a Web browser, specify the following URL:

   ```
   http://MachineName/ContextRoot/servlet/mstrWeb
   ```

   Alternatively, you can use the IP address of your machine for remote access, as shown below:

   ```
   http://IPAddress/ContextRoot/servlet/mstrWeb
   ```

   In these addresses, *ContextRoot* is the name you provided for the context root for Web Module box on Preparing for the application installation page. For example, the default name of the WAR file, which is MicroStrategy. For more information, refer to *Preparing for the application installation, page 238*.

You can now access the MicroStrategy Web Universal application.

**Uninstalling MicroStrategy Web application**

You can uninstall the MicroStrategy Web Universal application through the WebSphere Administrative console.
**To uninstall MicroStrategy Web application**

1. Expand **Applications**.
2. Click **Enterprise Applications**. A list of installed applications is displayed in the table to the right.
3. Select the desired MicroStrategy Web application.
4. Click **Uninstall**.
5. Save the configuration in the master repository.

**Performance-based setup information**

The performance of MicroStrategy Web Universal can be increased by configuring it on various component levels. These additional settings are not necessary, but can increase the performance of MicroStrategy Web Universal. This section explains the following changes:

- *Setting the Java heap size*
- *Precompiling JSP files*
- *Configuring the IBM HTTP Server to serve static files*

**Setting the Java heap size**

You can increase the Java heap size for a given application server by configuring the WebSphere Administrative Console.

**To increase the Java heap size**

1. Access the Administrative Console.
2. Expand the Servers node.
3. Click the **Application Servers** link to view the list of application servers.
4 Click the application server name, scroll to Additional Properties and click **Process Definition**.

5 Click **JVM Settings** to set the Java heap size settings. MicroStrategy recommends that you initially set the Java heap size to a minimum of 500MB, assuming the machine has enough memory space.

6 Click **Apply** and save your changes.

7 Stop and start the application server.

**Precompiling JSP files**

To avoid the time taken to load the Web pages in the application server when you access it for the first time, you must precompile the Java Server Pages (JSP) files. Precompilation can be done during deployment by selecting the **Enable pre-compile of JSPs** setting. Otherwise, it can be done after deploying the application.

To precompile the JSPs after deployment, set the application server to load all the pages in the application. Then when you connect for the first time, the pages are already loaded and performance is improved.

Before you precompile the JSP files, make sure that:

- The MicroStrategy Web Universal application is deployed in the WebSphere environment.
- You know the defined application name and the Web Module’s name. You can retrieve these names from the Administrative Console. Locate the application name under the Enterprise Applications node. Locate the Web Module name by expanding the application and clicking **Web Modules**. The default name is **Web Tier**.

---

**To precompile the JSP files**

1 Change the directory to **WAS_ROOT/bin**.

2 Run the following command:

```
./JspBatchCompiler.sh -enterpriseapp.name ApplicationName -webmodule.name webModule -cell.name cellName -node.name nodeName -server.name serverName -keepgenerated
```
TRUE

• If the administrative server is running in a security enabled mode, you are prompted for the user ID and password to connect to the Admin server.

username userID

password password

For each JSP file compiled without error, the following message appears:

Code generation successful.

Configuring the IBM HTTP Server to serve static files

The IBM HTTP Server (Web server) is tuned to effectively serve static files. As a result, perceived performance is greatly enhanced if you configure the IBM HTTP Server to serve image, style sheet, and JavaScript files. This also reduces the load on the WebSphere Server so that it can handle only dynamic files while IBM HTTP Server handles static files. This requires that you do the following:

• Configure the application server level to serve the Java Server Pages (JSPs) and servlets, which are dynamic files and handled by WebSphere.

• Configure the Web server level to serve the images, JavaScripts, and style sheets, which are static files and handled by the IBM HTTP Server.

For more information, see the IBM WebSphere Application Server redbook covering System Management and Configuration. This book discusses separating static content from dynamic content.

Deploying with Sun Java System (Solaris)

This section provides information used to deploy and configure MicroStrategy Web Universal on a Sun Java System Application Server 9.1 in a UNIX/Linux environment. You can also use the steps provided below to deploy MicroStrategy Mobile Server J2EE:

• Preconfiguration information: Configuration that must occur before you begin deploying MicroStrategy Web Universal.

• Deploying MicroStrategy Web Universal: Instructions for deploying.
• **Performance-based setup information**: Optional settings to increase the application’s performance.

These additional settings are not required, but can increase the performance of MicroStrategy Web Universal. Review this information prior to deployment to see if any of these options are of interest to you.


## Preconfiguration information

This section provides the preconfiguration information necessary for deploying MicroStrategy Web Universal on your machine. This includes:

• **Locating the WAR file**

It also provides additional setup information that needs to be done on the machine on which the application server and Web server are already installed.

## UNIX/Linux environment

The following table shows the recommended machine configuration to deploy MicroStrategy Web Universal with Sun Java System Application Server 9.1 in a UNIX/Linux environment:

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Solaris 9 or 10 (on SPARC)</td>
</tr>
<tr>
<td>Web server</td>
<td>Oracle iPlanet Web Server 6.1</td>
</tr>
<tr>
<td>Application server</td>
<td>Sun Java System Application Server 9.1</td>
</tr>
<tr>
<td>JDK</td>
<td>SUN JDK 1.5.0_12 (update 12)</td>
</tr>
</tbody>
</table>

You can download this JDK from the following location: [http://java.sun.com/j2se/1.5.0/download.html](http://java.sun.com/j2se/1.5.0/download.html)

For information on the version numbers certified and supported by MicroStrategy, see the MicroStrategy readme file.
Windows environment

The following table shows the recommended machine configuration for deploying MicroStrategy Web Universal in a Windows environment:

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Windows 2003 SP2 (on x86)</td>
</tr>
<tr>
<td>Web server</td>
<td>Oracle iPlanet Web Server 6.1</td>
</tr>
<tr>
<td>Application server</td>
<td>Sun Java System Application Server 9.1</td>
</tr>
<tr>
<td>JDK</td>
<td>SUN JDK 1.5.0_12 (update 12)</td>
</tr>
<tr>
<td></td>
<td>You can download this JDK from the following location:</td>
</tr>
<tr>
<td></td>
<td><a href="http://java.sun.com/j2se/1.5.0/download.html">http://java.sun.com/j2se/1.5.0/download.html</a></td>
</tr>
</tbody>
</table>

For information on the version numbers certified and supported by MicroStrategy, see the MicroStrategy readme file.

Locating the WAR file

The MicroStrategy Web Universal application is packaged within a single file, called a WAR file (Web ARchive), following the J2EE specifications. You must deploy this WAR file to run the application in your application server environment.

The MicroStrategy Web Universal WAR file (MicroStrategy.war) is located in the MicroStrategy Web Universal deployment directory you specified during installation. The default location when installing on 32-bit Windows environments is C:\Program Files\MicroStrategy\WebJSP. The default location when installing on 64-bit Windows environments is C:\Program Files (x86)\MicroStrategy\WebJSP. The default location when installing on UNIX or Linux is described in Directory structure, page 127.

To deploy the WAR file, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this chapter.
After deploying the MicroStrategy.war file, you can view the WEB-INF/log folder. The log folder retains all the log files except for the ncsportal.log files, which are created in the application server root folder. For more information on the directory structure after deploying the WAR file, see Directory structure after deploying the WAR file, page 215.

Locating the Mobile Server WAR file

If you are deploying MicroStrategy Mobile Server J2EE, you must use the WAR file for Mobile Server J2EE. The WAR file is placed in the folder you specified when installing Mobile Server J2EE. You can then use the same processes used to deploy the MicroStrategy Web Universal WAR file to deploy the Mobile Server J2EE WAR file.

Deploying MicroStrategy Web Universal

Once your machine has been configured, you can deploy MicroStrategy Web Universal with Sun Java System Application Server 9.1.

The Performance-based setup information section provides information on additional settings that can increase application performance. These additional settings are not required but can increase the performance of MicroStrategy Web Universal. Review this information prior to deployment to see if any of these settings are of interest to you.

The deployment involves the steps below, which are explained in detail in the following sections:

1. Launching the Sun Java System Application Server Admin Console
2. Deploying MicroStrategy Web Universal
3. Accessing the MicroStrategy Web Administrator page
4. Connecting to the project page

The administration and deployment tools in Sun Java System Application Server have the same interface regardless of the operating system on which they run. Therefore, the deployment process is the same for all operating systems, and is described below. There are some minor changes in the Windows environment, which are highlighted where necessary.
Launching the Sun Java System Application Server Admin Console

This procedure describes the steps to launch the Sun Java System Application Server.

To launch the Sun Java System Application Server Admin Console

1. If you do not have a domain created for your server, create a domain along with a user name and password for the domain. In a command prompt type the following command:

   `asadmin create-domain --adminuser UserName --adminport PortNumber --domaindir DomainDirectory DomainName`

   where:
   - `UserName` is a user name to use for the domain.
   - `PortNumber` is the port number to use for the administration server.
   - `DomainDirectory` is the path for the domain. You can remove the `--domaindir` option to use the default domain directory.
   - `DomainName` is the name to distinguish the domain.

2. Press ENTER. You are prompted to provide a new password. Type a password.

3. Navigate to the following directory in the command prompt:

   `InstallDir/bin`

4. Type the following command to start the domain:

   `asadmin start-domain --domaindir DomainDirectory DomainName`

   where:
   - `DomainDirectory` is the path you defined when creating the domain. You can remove the `--domaindir` option if the domain uses the default directory.
   - `DomainName` is the name of the domain you created in the previous steps.
5 Once the domain is started successfully, access the Sun Java System Application Server Admin Console by typing the following URL:

http://MachineName:PortNumber

where:

- **MachineName** is the IP address or the name of the machine where you installed Sun Java System Application Server.
- **PortNumber** is the port number you provided when creating the domain.

In a Windows environment, from the Start menu, point to **Programs**, then **Sun Microsystems**, then **Application Server 9.1**, and then choose **Admin Console**.

6 Type the user name and password that you provided when creating the domain.

### Deploying MicroStrategy Web Universal

After launching the Sun Java System Application Server Admin Console, follow the steps below to deploy MicroStrategy Web Universal as a WAR file. The steps provided below can also be used to deploy MicroStrategy Mobile Server J2EE as a WAR file.

---

**To deploy MicroStrategy Web Universal as a WAR file**

1 Access the Admin Console, using the user name and password that you used to create the domain. The steps to access the Admin Console are provided in *Launching the Sun Java System Application Server Admin Console, page 250*.

2 Expand the **Application Server** tree view.

3 Expand **Applications** under the server you are using. By default, the application server created is **server1**.

4 Select **Web Applications** and click **Deploy**. Deploy Web Module opens in the right pane.

5 Type the full path or browse to the location where you want to deploy the WAR file.
If you are deploying MicroStrategy Mobile Server J2EE instead, use the MicroStrategyMobile.war file for the deployment process.

6 Select or clear the following check boxes according to your requirements:

- Re-deploy if already exists
- Run Verifier
- Precompile JSPs

![It is recommended you select the Precompile JSPs check box to quickly load the Web pages in the application server when you access it for the first time.]

- Virtual Servers

7 Click OK.

Controlling access to the MicroStrategy Web Administrator page

For security purposes, you must only assign certain users the authorization to access the MicroStrategy Web Administrator page. To do this, map the Admin security role to only those users for whom you want to grant permission to work in the MicroStrategy Web Administrator page.

Sun Java System Application Server supports the following authentication realms:

- File
- Admin-realm
- Certificate

A realm, also called a security policy domain or security domain, is a scope over which a common security policy is defined and enforced by the security administrator of the security service. For more information, see the following resource:

http://docs.sun.com/app/docs/doc/819-3671/abloe?azview

In Sun Java System Application Server, the file realm is selected by default. To create users in the default realm, click File, then click Manage Users to start creating users authorized to access the MicroStrategy Web Administrator page.
Mapping a security role

A descriptor file, `sun-web.xml`, enables Sun Java System Application Server to map the existing users or groups to different security roles. This file is located in the `WebApplicationRootDir/WEB-INF` folder. You can modify this file if you want to change the default values for security roles.

The contents of this file are as follows:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sun-web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Application Server 8.1 Servlet 2.4//EN" "http://www.sun.com/software/appserver/dtds/sun-web-app_2_4-1.dtd">
<sun-web-app xmlns="http://java.sun.com/xml/ns/j2ee">
  <context-root>/MicroStrategy</context-root>
  <security-role-mapping>
    <role-name>admin</role-name>
    <principal-name>admin</principal-name>
  </security-role-mapping>
</sun-web-app>
```

For mapping security roles to users and user groups, the `sun-web.xml` file assumes the default values that are provided during installation of Sun Java System Application Server and deployment of MicroStrategy Web Universal. The following are the values taken for context-root, role mapping, and the user or principal user:

- Context-root is set to MicroStrategy. This is the default value for the context name when deploying MicroStrategy Web Universal.
- Role mapping is set to admin. This is the default value specified in the web.xml file.
- The user or principal name is also set to admin. This is the default name assigned to the user when installing Sun Java System Application Server.

If you make changes to the default values of context, role, or user name while installing Sun Java System Application Server or
deploying MicroStrategy Web Universal, you must update the

**Accessing the MicroStrategy Web Administrator page**

When installing Sun Java System Application Server, the user that you create
is saved in the admin realm. By default, the user name is admin. You must
enable this admin user to access the MicroStrategy Web Administrator page.
The steps provided below can also be used to enable access to the
MicroStrategy Mobile Server Administrator page.

To access the admin user, you must change the default realm, file, to the
admin realm.

---

**To access the MicroStrategy Web Administrator page**

1. Expand Configuration from the tree view, and select Security. Security
   features are displayed on the right pane.

2. Select the admin realm from the Default Realm drop-down list.

3. Click Save to change the default realm.

4. Stop and restart the application server. For more information, see
   *Restarting the application server, page 254*.

**Restarting the application server**

You must stop and restart the application server instance to apply changes.
This is necessary after most changes to the application server configuration,
such as creating a user.

---

**To restart the application server**

After changing the application server configuration or creating a user, a
message Apply Changes Required is displayed.

1. Click on the message Apply Changes Required.
If the server must be restarted, the Restart Required message is displayed.

2 Click **Restart**.

You can also restart the application server by navigating through the left menu, as follows:

1 Expand Application Server.

2 Select the server instance you are currently working on and click the **Stop Instance** button displayed on the right panel. The Sun Java System Application Server Administration Console window opens.

3 Follow the instructions given in the Sun Java System Application Server Administration Console window to stop and restart the application server.

After restarting the application server, follow the procedure below to connect to the project page.

**Connecting to the project page**

After restarting the application server, follow the steps described here to connect to the project page.

**To connect to the Project page**

In a Web browser, type the following URL:

http://machine_name:HTTP_port/context

If you have used all the default variables, you can access the following URLs:

http://localhost:80/MicroStrategy/

or

http://localhost:80/MicroStrategy/servlet/mstrWeb

You are now ready to use MicroStrategy Web Universal.
Performance-based setup information

The performance of MicroStrategy Web Universal can be increased by configuring various components. These additional settings are not necessary, but they can improve the performance of MicroStrategy Web Universal. This section provides the following configurations:

- Setting the Java heap size
- Undeploying MicroStrategy Web Universal, after deployment

Setting the Java heap size

The maximum Java heap size, which handles all the Java requests, is set to 512MB by default.

To set the Java heap size

1. Navigate to **Application server**.

2. Click **JVM Settings** and select **JVM Options**. You can view the argument `-Xmx512`. This is the default value for the maximum heap size.

3. Click **Add JVM Options** to include additional arguments using `-Xmx -Xms`. For example, if you want to set the maximum heap size to 1GB and minimum heap size to 512MB, then the argument you provide is `-Xmx1024m and -Xms512m`.

Undeploying MicroStrategy Web Universal

Oracle recommends undeploying an application before deploying a newer version. The steps below show you how to undeploy an existing MicroStrategy Web Universal application, using the Sun Java System Application Server Admin Console.

To undeploy MicroStrategy Web Universal

1. Access the Admin Console.
2 Type the user name and password that you assigned during the installation. By default, the user name is admin.

3 Expand the Application Server tree view.

4 Expand Applications under the server you are using. By default, the application server created is server1.

5 Select Web Applications.

6 Select the check box for the web application you want to undeploy and click Undeploy.

7 After the undeployment is finished, stop and restart the application server for the changes to take effect.

## Deploying with Tomcat (Windows)

This section provides information used to deploy and configure MicroStrategy Web Universal in a Tomcat-only environment. For information on how to configure Tomcat to work with IIS, see iishowto.html (Tomcat 6.0) in the Tomcat documentation. You can also use the steps provided below to deploy MicroStrategy Mobile Server J2EE:

- **Preconfiguration information**: Configuration that must occur before you begin deploying MicroStrategy Web Universal.

- **Deploying MicroStrategy Web Universal**: Instructions for deploying, including step-by-step procedures.

### Preconfiguration information

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Universal on Tomcat on your machine.
While your setup may vary slightly, for example, you may have different versions of these applications, the overall process remains the same.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Windows 2003 SP2 (on x86)</td>
</tr>
<tr>
<td>Application server</td>
<td>Tomcat 6.0</td>
</tr>
</tbody>
</table>
| JDK             | SUN JDK 1.5.0  
You can download this JDK from the following location:  
http://java.sun.com/j2se/1.5.0/download.html |

**Configuring the JDK**

If you have not installed SUN JDK 1.5.0 yet, download the file from the http://java.sun.com/j2se/1.5.0/download.html website. Be sure to install the JDK, not the JRE.

The download site displays a number of software options. These might include terms such as JRE, JDK, and Java SDK. You must install a development kit (which is termed JDK or SDK) rather than installing only the JRE.

To configure the JDK, a system variable must point to the folder where you install the JDK. If you install the JDK to a simple folder path such as C:\ setting the system variable is easier and more likely to be correct.

After you install the Sun JDK, you must configure it.

---

**To configure the JDK**

1. On your Windows Desktop, right-click **My Computer** and select **Properties**. The System Properties dialog box opens.

2. Select the **Advanced** tab.

3. Click **Environment Variables**. The Environment Variables dialog box opens.

4. Under **System Variables**, click **New** to create a system variable. The New System Variable dialog box opens.
5 In the **Variable Name** box, type **JAVA_HOME**.

6 In the **Variable Value** box, specify the path of the folder where you installed the JDK and click **OK**.

For example, if the fully qualified path to your JDK executable is `C:\jdk1.5.0_12\bin\java.exe`, the value of your **JAVA_HOME** variable is `C:\jdk1.5.0_12`.

If you have installed JDK under the **Program Files** folder, type `Progra~1` when specifying the folder name in the **Variable Value** box; otherwise the system does not recognize the folder. For example, type `C:\Progra~1\jdk1.5.0` in the **Variable Value** box.

### Configuring Tomcat

This procedure assumes that you have downloaded and installed Tomcat on your machine. You can download Tomcat from the Apache website; depending on the version you want to download, you may need to locate the appropriate file in Apache’s Archive area. Instructions for downloading and installing Tomcat are also available on the Apache website.

To configure Tomcat, a system variable must point to the folder where you install Tomcat. If you install Tomcat to a simple folder path such as `C:\Tomcat` then setting the system variable is easier and more likely to be correct.

After you install Tomcat, you must configure it.

---

**To configure Tomcat**

1 On your Windows Desktop, right-click **My Computer** and select **Properties**. The System Properties dialog box opens.

2 Select the **Advanced** tab.

3 Click **Environment Variables**. The Environment Variables dialog box opens.

4 Under **System Variables**, click **New** to create a system variable. The New System Variable dialog box opens.

5 In the **Variable Name** box, type **CATALINA_HOME**.
In the **Variable Value** box, specify the path of the folder where you installed Tomcat and click **OK**. For example, if you installed Tomcat directly to the C drive, the destination folder is `C:\Tomcat`.

If you installed Tomcat under the `Program Files` folder, type `Progra~1` when specifying the folder in the Variable Value box. Otherwise, the system does not recognize the folder. For example, type `C:\Progra~1\Tomcat` in the Variable Value box.

**Locating the WAR file**

The MicroStrategy Web Universal application is packaged within a single file, called a WAR file (Web ARchive), following the J2EE specifications, which you must deploy to run the application in your application server environment.

The MicroStrategy Web Universal WAR file (`MicroStrategy.war`) is located in the path you specified when installing MicroStrategy Web Universal. The default location when installing on 32-bit Windows environments is `C:\Program Files\MicroStrategy\WebJSP`. The default location when installing on 64-bit Windows environments is `C:\Program Files (x86)\MicroStrategy\WebJSP`. The default location when installing on UNIX or Linux is described in *Directory structure, page 127*.

To deploy the WAR file, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this chapter.

**Locating the Mobile Server WAR file**

If you are deploying MicroStrategy Mobile Server J2EE, you must use the WAR file for Mobile Server J2EE. The WAR file is placed in the folder you specified when installing Mobile Server J2EE. You can then use the same processes used to deploy the MicroStrategy Web Universal WAR file to deploy the Mobile Server J2EE WAR file.
Deploying MicroStrategy Web Universal

Assuming you have made all the necessary configurations described above, you can begin deploying MicroStrategy Web Universal with Tomcat. This involves the following steps:

1. **Deploying using Tomcat as a stand-alone Web container**
2. **Controlling access to the MicroStrategy Web Administrator page**
3. **Accessing the MicroStrategy Web Administrator page**
4. **Launching the project**

**Deploying using Tomcat as a stand-alone Web container**

To deploy MicroStrategy Web Universal using Tomcat as a stand-alone Web container

1. Copy the `MicroStrategy.war` file to the `Tomcat\webapps` folder. Instructions on locating the MicroStrategy.war file can be found in the section *Locating the WAR file, page 260.*

   If you are deploying MicroStrategy Mobile Server J2EE instead, locate the `MicroStrategyMobile.war` file (see *Locating the Mobile Server WAR file, page 260*).

2. **Stop and start Tomcat from the command line**

   2. From the Start menu, select Run. The Run dialog box opens.

   3. Type `cmd` in the Open drop-down list and click OK. The command prompt opens.

   4. Browse to the `Tomcat\bin` folder, where `Tomcat` is the folder in which you installed Tomcat. For example, in the command prompt, type

      ```
      cd C:\Tomcat\bin
      ```

   5. Press ENTER.

      `C:\Tomcat\bin>` is displayed at the command prompt.
6 Type `startup` to start Tomcat and `shutdown` to stop it.

If you installed Tomcat under the `Program Files` folder, type `Progra~1` when you change folders in the command prompt. Otherwise, the system does not recognize the folder. For example, type `C:\Progra~1\Tomcat\bin` in the command prompt.

MicroStrategy Web Universal is deployed automatically. There is now a MicroStrategy folder under `Tomcat\webapps` folder. If you deployed MicroStrategy Mobile Server J2EE instead, the folder is named MicroStrategyMobile.

`ncsportal.log` files are created in the Application server root folder and not in the `WEB-INF\log` folder of MicroStrategy.

**Controlling access to the MicroStrategy Web Administrator page**

The MicroStrategy Web Administrator page is accessible only to users with an “admin” role. To create a user/password set that is authorized to access the Administrator page, you must create the users and assign them the role of admin under the Tomcat user configuration file. The steps to give a user access to the MicroStrategy Web Administrator page follow. If you are deploying Mobile Server J2EE, you can use the steps below to control access to the MicroStrategy Mobile Server Administrator page.

**To control access to the MicroStrategy Web Administrator page**

1. In the `Tomcat\conf` folder, open the `tomcat-users.xml` file in a program that allows you to edit the file, such as Notepad.

2. Add the following tag and save the file:

   ```xml
   <user name="administrator" password="administrator"
   roles="admin"/>
   ```

   You can specify any value in the `user name` and `password` fields. These are used to log in to the MicroStrategy Web Administrator page.

3. Stop and start Tomcat from the command line.
Now you can start MicroStrategy Web Administrator to add and connect to an Intelligence Server. To configure MicroStrategy Web Administrator, see Accessing the MicroStrategy Web Administrator page, page 269.

## Accessing the MicroStrategy Web Administrator page

**To configure the MicroStrategy Web Administrator page**

1. In a Web browser, access the MicroStrategy Web Administrator page or the MicroStrategy Mobile Server Administrator page by specifying the following URL:
   - For Web: `http://localhost:8080/MicroStrategy/servlet/mstrWebAdmin`
   - For Mobile Server J2EE: `http://localhost:8080/MicroStrategy/servlet/MobileServlet`

   The servlet names are case-sensitive. Make sure to use the correct case when typing the `mstrWebAdmin` or `MobileServlet` name. If the application server is enabled with security, a dialog box related to the administrator authentication opens.

   If you are using Tomcat integrated with IIS, you do not need to specify the port number in the URL. However, when using Tomcat as a stand-alone Web container, you must specify the port number. The default port for Tomcat is 8080.

2. When prompted for a user name and password, use the same values you specified in the `tomcat-users.xml` file.

3. Once you log in, the MicroStrategy Web Administrator page is displayed. Add and connect to an Intelligence Server.

   If you are deploying MicroStrategy Mobile Server J2EE, after you are authenticated, the MicroStrategy Mobile Server Administrator page appears. Add and connect to an Intelligence Server. Once connected, click Mobile Configuration to configure your MicroStrategy Mobile applications to communicate with Mobile Server and Intelligence Server. For steps on how to define this configuration, see the MicroStrategy Mobile Design and Administration Guide. Creating a configuration completes the steps required to deploy Mobile Server J2EE.
Deploying with Tomcat (Linux)

This section provides information on how to deploy and configure MicroStrategy Web Universal with Tomcat in a Linux environment. You can also use the steps provided below to deploy MicroStrategy Mobile Server J2EE:

• **Preconfiguration information**: Configuration that must occur before you begin deploying MicroStrategy Web Universal

• **Deploying MicroStrategy Web Universal**: Instructions for deploying, including detailed steps

Preconfiguration information

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Universal with Tomcat on your Linux machine:

• **Installing the JDK**

• **Configuring the JDK**

• **Installing Tomcat**

• **Configuring Tomcat**

• **Locating the WAR file**
While your setup may vary slightly, for example, you may have different versions of these applications, the overall process remains the same.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>RedHat Enterprise Linux AS Version 4 update 6 (on x86-64)</td>
</tr>
<tr>
<td>Application server</td>
<td>Tomcat 6.0</td>
</tr>
</tbody>
</table>
| JDK | SUN JDK 1.5.0  
You can download this JDK from the following location: [http://java.sun.com/j2se/1.5.0/download.html](http://java.sun.com/j2se/1.5.0/download.html) |

### Installing the JDK

If you have not installed SUN JDK 1.5.0 yet, download the shell file from the [http://java.sun.com/j2se/1.5.0/download.html](http://java.sun.com/j2se/1.5.0/download.html) website. Be sure to install the JDK, not the JRE.

The download site displays a number of software options. These might include terms such as JRE, JDK, and Java SDK. You must install a development kit (which is termed JDK or SDK) rather than installing only the JRE.

To configure the JDK, a system variable must point to the folder where you install the JDK. If you install the JDK to a simple folder path such as `C:\`, setting the system variable is easier and more likely to be correct.

From the location in which to install the JDK, run the file you downloaded:

```
jdk-\textit{Version}-linux-i586.bin
```

For example, to install version 1.5.0, type the following:

```
jdk-1_5_0_12-linux-i586.bin
```

### Configuring the JDK

After you install the Sun JDK, you must configure it.
To configure the JDK

1. Open the `/etc/profile` file using a program that allows you to edit the file.

2. Add the following line:

   ```sh
   export JAVA_HOME=/PathName/jdkVersion;
   ```

   where `PathName` is the destination folder where you installed the JDK and `Version` is the version, such as `1_5_0_12`, of the JDK.

Installing Tomcat

This procedure assumes that you have downloaded and installed Tomcat in a directory named `Tomcat` on your machine. If you have not installed Tomcat 6.0 yet, download it from the `http://tomcat.apache.org/download-60.cgi` website. Retrieve the zip file for your version of Tomcat 6.0.

Contact your System Administrator or visit the Apache website for instructions on downloading and installing Tomcat.

Configuring Tomcat

After you install Tomcat, you must configure Tomcat. The Tomcat configuration includes creating the environment variable `CATALINA_HOME` and defining this environment variable to point to the Tomcat directory.

To configure Tomcat

1. Open the `etc/profile` file in a program that allows you to edit the file.

2. Type the following:

   ```sh
   export CATALINA_HOME = /PathName
   ```

   where `PathName` is the directory where you have installed Tomcat.

   For example,

   ```sh
   export CATALINA_HOME = /Tomcat
   ```
Locating the WAR file

The MicroStrategy Web Universal application is packaged within a single file, called a WAR file (Web ARchive), following the J2EE specifications. You must deploy the WAR file to run the application in your application server environment.

The MicroStrategy Web Universal WAR file (MicroStrategy.war) is located in the path you specified when installing MicroStrategy Web Universal. The default location when installing on 32-bit Windows environments is C:\Program Files\MicroStrategy\WebJSP. The default location when installing on 64-bit Windows environments is C:\Program Files (x86)\MicroStrategy\WebJSP. The default location when installing on UNIX or Linux is described in Directory structure, page 127.

To deploy the MicroStrategy.war file, you must follow a set of steps that are specific to the application server you are using. For more details, see your application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this chapter.

After deploying the MicroStrategy.war file, you can view the WEB-INF folder, which contains a subfolder named log. The log folder retains all the log files except for the ncsportal.log files. They are created in the Application server root folder. For more information on the directory structure after deploying the WAR file, see Directory structure after deploying the WAR file, page 215.

Locating the Mobile Server WAR file

If you are deploying MicroStrategy Mobile Server J2EE, you must use the WAR file for Mobile Server J2EE. The WAR file is placed in the folder you specified when installing Mobile Server J2EE. You can then use the same processes used to deploy the MicroStrategy Web Universal WAR file to deploy the Mobile Server J2EE WAR file.
Deploying MicroStrategy Web Universal

After you have performed the configurations described above, you can begin deploying MicroStrategy Web Universal with Tomcat. This involves the following steps:

1. **Deploying using Tomcat as a standalone Web container**
   - Copy the `MicroStrategy.war` file to the `Tomcat/webapps` directory. Instructions on locating the `MicroStrategy.war` file can be found in the previous section, *Locating the WAR file, page 267.*
   - If you are deploying MicroStrategy Mobile Server J2EE instead, copy the `MicroStrategyMobile.war` file to the `Tomcat/webapps` directory (see *Locating the Mobile Server WAR file, page 267*). Use this WAR file for the rest of the steps in this procedure.

   **To start and stop Tomcat from the command line**

   2. Type `# $CATALINA_HOME/bin/startup.sh` and press ENTER to start Tomcat, which deploys MicroStrategy Web Universal automatically. There is now a `MicroStrategy` folder under the `Tomcat/webapps` directory. If you deployed MicroStrategy Mobile Server J2EE instead, the folder is named `MicroStrategyMobile`.

Controlling access to the MicroStrategy Web Administrator page

The MicroStrategy Web Administrator page, `mstrWebAdmin` servlet, is accessible only to users with an admin role. To create a user and password set that is authorized to access the Administrator page, you must create the
users and assign them the admin role under the Tomcat user configuration file. If you are deploying Mobile Server J2EE, you can use the steps below to control access to the MicroStrategy Mobile Server Administrator page.

**To control access to the MicroStrategy Web Administrator page**

1. In the `Tomcat/conf` directory, open the `tomcat-users.xml` file using a program that allows you to edit the file.

2. Add the following tags and save the file:

   ```xml
   <role rolename="admin"/>
   <user username="admin" password="admin" roles="admin"/>
   <user username="system" password="goldengate" roles="admin"/>
   ```

   You can specify any value in the `username` and `password` fields. These are used to log in to the MicroStrategy Web Administrator page. The `roles` field must be `admin`.

3. Stop and restart Tomcat.

Now you can start MicroStrategy Web Administrator to add and connect to an Intelligence Server. To configure MicroStrategy Web Administrator, see *Accessing the MicroStrategy Web Administrator page*, page 269.

**Accessing the MicroStrategy Web Administrator page**

**To configure the MicroStrategy Web Administrator page**

1. In a Web browser, access the MicroStrategy Web Administrator page or the MicroStrategy Mobile Server Administrator page by specifying the following URL:

   - **For Web**: `http://localhost:8080/MicroStrategy/servlet/mstrWebAdmin`
   - **For Mobile Server J2EE**: `http://localhost:8080/MicroStrategy/servlet/MobileServlet`
The servlet names are case-sensitive. Be sure to use the correct case when typing the `mstrWebAdmin` or `MobileServlet` name. If the application server is enabled with security, a dialog box related to the administrator authentication opens.

2 When prompted for a user name and password, use the same values you specified in the `tomcat-users.xml` file.

3 When you log in, the MicroStrategy Web Administrator page is displayed. Add and connect to an Intelligence Server.

   If you are deploying MicroStrategy Mobile Server J2EE, after you are authenticated, the MicroStrategy Mobile Server Administrator page appears. Add and connect to an Intelligence Server. Once connected, click Mobile Configuration to configure your MicroStrategy Mobile applications to communicate with Mobile Server and Intelligence Server. For steps on how to define this configuration, see the *MicroStrategy Mobile Design and Administration Guide*. Creating a configuration completes the steps required to deploy Mobile Server J2EE.

4 Proceed to launch the MicroStrategy Web Universal project. For more information, see *Launching the project, page 270*.

### Launching the project

In a Web browser, access MicroStrategy Web Universal using the following URL:

http://localhost:8080/MicroStrategy/servlet/mstrWeb

### Deploying with SAP NetWeaver (Windows)

This section provides information used to deploy and configure MicroStrategy Web Universal on a Windows machine using the SAP application server. You can also use the steps provided below to deploy MicroStrategy Mobile Server J2EE:

- **Preconfiguration information**: Configuration that must occur before you begin deploying MicroStrategy Web Universal.

  - **Deploying MicroStrategy Web Universal**: Instructions for deploying the application.
Preconfiguration information

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Universal on your machine. It also provides additional setup information for the machine where the application server and Web server are already installed.

The configuration required to deploy MicroStrategy Web Universal on SAP is provided in the following table. While your configuration may vary slightly, for example, you may have different versions of these applications, the overall process remains the same.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Windows 2003 SP2 (on x86)</td>
</tr>
<tr>
<td>Application server</td>
<td>SAP NetWeaver Application Server 7.1</td>
</tr>
</tbody>
</table>

For information on the version numbers certified and supported by MicroStrategy, see the MicroStrategy readme file.

Locating the WAR file

The MicroStrategy Web Universal application is packaged within a single file, called a WAR file (Web ARchive), following the J2EE specifications. You must deploy this WAR file to run the application in your application server environment.

The MicroStrategy Web Universal WAR file (MicroStrategy.war) is located in the path you specified when installing MicroStrategy Web Universal. The default location when installing on 32-bit Windows environments is C:\Program Files\MicroStrategy\WebJSP. The default location when installing on 64-bit Windows environments is C:\Program Files (x86)\MicroStrategy\WebJSP. The default location when installing on UNIX or Linux is described in Directory structure, page 127.

To deploy the WAR file, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this chapter.
After deploying the `MicroStrategy.war` file, you can view the `WEB-INF/log` folder. The `log` folder retains all the log files. For more information on the directory structure after deploying the WAR file, see *Directory structure after deploying the WAR file, page 215*.

**Locating the Mobile Server WAR file**

If you are deploying MicroStrategy Mobile Server J2EE, you must use the WAR file for Mobile Server J2EE. The WAR file is placed in the folder you specified when installing Mobile Server J2EE. You can then use the same processes used to deploy the MicroStrategy Web Universal WAR file to deploy the Mobile Server J2EE WAR file.

**Deploying MicroStrategy Web Universal**

Once your machine has the necessary settings configured, you can deploy MicroStrategy Web Universal on the SAP-Windows machine. Deployment involves the following steps:

1. **Deploying MicroStrategy Web Universal with the SAP NetWeaver Application Server**
2. **Controlling access to the MicroStrategy Web Administrator Page**
3. **Accessing the MicroStrategy Web Administrator Page**

**Deploying MicroStrategy Web Universal with the SAP NetWeaver Application Server**

Follow the steps provided in this section to deploy MicroStrategy Web Universal as a WAR file.

---

**To deploy MicroStrategy Web Universal as a WAR file**

1. Copy the `MicroStrategy.war` file to the Windows machine hosting your application server. The location in which you store the file is used later and referred to as `path_to_war_file`.
If you are deploying MicroStrategy Mobile Server J2EE instead, copy the MicroStrategyMobile.war file to the Windows machine hosting your application server. Use this WAR file for the rest of the steps in this procedure.

2 From the Windows Start menu, select Run. The Run dialog box opens.

3 In the Open drop-down list, type cmd, and click OK. A command prompt opens.

4 Using the command prompt, browse to the following directory within the SAP Application Server installation directory:

/usr/sap/SID/Instance_Number/j2ee/deployment/scripts/

The SID and Instance_Number parameters are defined during installation and configuration of the SAP Application Server. The port number above refers to the P4 port number. The default port number is 50004.

5 Type the following command and press ENTER to deploy the MicroStrategy.war file:

Deploy.bat

user_name:password@localhost:port_number
path_to_war_file

The user name and password must have administrative access. The port number above refers to the P4 port number. The default port number is 50004.

6 Access NetWeaver web admin console using the following URL:

http://localhost:PortNumber/nwa

The PortNumber above refers to the J2EE engine port number. The default port number is 50000.

7 Login as an administrative user.

8 From the Operation Management tab, select the Systems tab, and then click Start & Stop.

9 Select Java EE Applications. A list of applications deployed on the application server are displayed.

10 Select the MicroStrategy application just deployed from the list.
11 From the Application Details section, select the Status tab, and then click Start.

12 Select On all instances and Set "Started" as Initial State.

When application is started, the status is displayed as Started.

Controlling access to the MicroStrategy Web Administrator Page

You can configure security roles to control access to the MicroStrategy Web Administrator page. A security role named “admin” is created for the MicroStrategy Web Universal application within SAP NetWeaver, and is mapped to the “administrators” user group. This is defined in the web-j2ee-engine.xml file. To allow access to the MicroStrategy Web Administrator page, modify this file to map users or groups to this admin security role, or include users in the administrators user group.

Accessing the MicroStrategy Web Administrator Page

Before you start MicroStrategy Web Universal, you must configure the MicroStrategy Web Administrator page. If you are deploying Mobile Server J2EE, you can use the steps below to configure the MicroStrategy Mobile Server Administrator page.

You must have administrative privileges to access the MicroStrategy Web Administrator page. For more information, see Controlling access to the MicroStrategy Web Administrator Page, page 274.

To access the MicroStrategy Web Administrator page

1 Access the servlet by typing the following URL in a Web browser:

- For Web: http://MachineName:PortNumber/MicroStrategy/servlet/mstrWebAdmin

- For Mobile Server J2EE: http://MachineName:PortNumber/MicroStrategy/servlet/MobileServlet

The servlet names are case-sensitive. Use the correct case when typing the mstrWebAdmin or MobileServlet name.

The login dialog box opens.
2 Specify a user name and password.

3 After you are authenticated, the MicroStrategy Web Administrator page opens. Use it to add and connect to an Intelligence Server.

If you are deploying MicroStrategy Mobile Server J2EE, after you are authenticated, the MicroStrategy Mobile Server Administrator page appears. Add and connect to an Intelligence Server. Once connected, click **Mobile Configuration** to configure your MicroStrategy Mobile applications to communicate with Mobile Server and Intelligence Server. For steps on how to define this configuration, see the *MicroStrategy Mobile Design and Administration Guide*. Creating a configuration completes the steps required to deploy Mobile Server J2EE.

4 Access the MicroStrategy Web Application on SAP Web Server by specifying the following URL in the Web browser:

   http://MachineName:PortNumber/MicroStrategy/servlet/mstrWeb

---

**Deploying with Oracle 10g (Windows)**

This chapter provides information used to deploy and configure MicroStrategy Web Universal on the Windows 2000 operating system with Apache as the Web server and Oracle 10g as the application server. You can also use the steps provided below to deploy MicroStrategy Mobile Server J2EE:

- **Preconfiguration information**: Configuration that must occur before you begin deploying MicroStrategy Web Universal.

- **Deploying MicroStrategy Web Universal**: Instructions for deploying, including detailed steps.

---

**Preconfiguration information**

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Universal on your machine.
This section uses the following setup. While your setup may vary slightly, for example, you may have different versions of these applications, the overall process remains the same.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Windows 2003 SP2 (on x86)</td>
</tr>
<tr>
<td>Application server</td>
<td>Oracle 10g 10.1.3</td>
</tr>
<tr>
<td>Web server</td>
<td>Apache 2.0.x</td>
</tr>
<tr>
<td>JDK</td>
<td>SUN JDK 1.5.0</td>
</tr>
</tbody>
</table>

You can download this JDK from the following location: [http://java.sun.com/j2se/1.5.0/download.html](http://java.sun.com/j2se/1.5.0/download.html)

### Locating the WAR file

The MicroStrategy Web Universal application is packaged within a single file, called a WAR file (Web ARchive), following the J2EE specifications, which you must deploy to run the application in your application server environment.

The MicroStrategy Web Universal WAR file (MicroStrategy.war) is located in the path you specified when installing MicroStrategy Web Universal. The default location when installing on 32-bit Windows environments is `C:\Program Files\MicroStrategy\WebJSP`. The default location when installing on 64-bit Windows environments is `C:\Program Files (x86)\MicroStrategy\WebJSP`. The default location when installing on UNIX or Linux is described in *Directory structure, page 127.*

To deploy the MicroStrategy.war file, you must follow a set of steps that are specific to the application server you are using. For more details, see your application server vendor documentation or follow the instructions within this guide.

After deploying the MicroStrategy.war file, you can view the WEB-INF folder, which contains a subfolder named log. The log folder retains all the log files except for the ncsportal.log files. They are created in the application server root folder. For more information on the directory structure after deploying the WAR file, see *Directory structure after deploying the WAR file, page 215.*
Locating the Mobile Server WAR file

If you are deploying MicroStrategy Mobile Server J2EE, you must use the WAR file for Mobile Server J2EE. The WAR file is placed in the folder you specified when installing Mobile Server J2EE. You can then use the same processes used to deploy the MicroStrategy Web Universal WAR file to deploy the Mobile Server J2EE WAR file.

Deploying MicroStrategy Web Universal

After your machine is configured, you can start the deployment of MicroStrategy Web Universal with Oracle 10g.

To deploy MicroStrategy Web Universal, perform the following procedures:

1. **Deploying using the Oracle Enterprise Manager**
2. **Accessing the MicroStrategy Web Administrator page**
3. **Launching the project**

Deploying using the Oracle Enterprise Manager

You can access Oracle Enterprise Manager from the following URL:

http://MachineName:PortNumber/em

where *MachineName* is the machine name or IP address of the Oracle machine, and *PortNumber* is the port number of Oracle Enterprise Manager.

To deploy with the Oracle Enterprise Manager

1. Start the Apache Web Server. From the **Start** menu, point to **OracleAS 10g - DEFAULT_HOME1**, and then choose **Start ApplicationServerName.MachineName.domain**.

2. To verify that the Apache Web Server has started, open Oracle Enterprise Manager, select **HTTP Server**, and then click **Start**.
3 Select the OC4J instance where you want to deploy MicroStrategy Web Universal. This procedure assumes you are using the default instance name home. Click home. The OC4J: home page opens.

4 Select the Applications tab.

5 Click Deploy. The Deploy: Select Archive page opens.

6 In the Archive area, select Archive is present on local host.

7 Click Browse to navigate to and select the MicroStrategy.war file.

If you are deploying MicroStrategy Mobile Server J2EE instead, locate the MicroStrategyMobile.war file. Use this WAR file for the rest of the steps in this procedure.

For more information on locating the WAR file, see Locating the WAR file, page 276.

8 In the Deployment Plan area, select Automatically create a new deployment plan and click Next. The Deploy: Application Attributes page opens.

9 Enter the Application Name and Context Root. This section on deploying MicroStrategy Web Universal with Oracle 10g uses MicroStrategy as the Application Name and /MicroStrategy as the Context Root.

10 Click Next. The Deploy: Deployment Settings page opens.

To map a user to the admin security role

The MicroStrategy Web Administrator page (mstrWebAdmin servlet) is accessible only to users with an admin security role. To control access to it, map the admin security role only to those users for whom you want to grant permission to work in the MicroStrategy Web Administrator page. In Oracle 10g, the security users and groups are defined in the Oracle Enterprise Manager.

If you are deploying MicroStrategy Mobile Server J2EE instead, you can use the same steps provided below to control access to the MicroStrategy Mobile Server Administrator page (MobileServlet servlet).

11 In the Map Security Roles task name, click the Go To Task (pencil) icon. The Deployment Settings: Map Security Roles page page opens.
For the **admin** security role, select the **Map Role** (pencil) icon. The Deployment Settings: Map Security Role: admin page opens.

Select **Map selected users and groups to this role**.

In the **Map Role to Users** area, in the **User** text box, type the user name to map to the admin security role and click **Add**.

Repeat this step to add all users for whom you want to grant permission to work in the MicroStrategy Web Administrator page.

**Click Continue**, and then click **OK**. You are returned to the Deploy: Deployment Settings page.

**Click on Deploy** to deploy the MicroStrategy Web application.

**Stop and restart the Apache Web Server.**

Any users mapped to the admin security role can now access the MicroStrategy Web Administrator page to add and connect to an Intelligence Server. For information about this, see **Accessing the MicroStrategy Web Administrator page** below.

**Accessing the MicroStrategy Web Administrator page**

1. In a Web browser, access the administrator page by specifying the following URL:
   - **For Web**: http://{IPAddress}:{PortNumber}/MicroStrategy/servlet/mstrWebAdmin
   - **For Mobile Server J2EE**: http://{IPAddress}:{PortNumber}/MicroStrategy/servlet/MobileServlet

   where **IPAddress** is the IP address of the Oracle machine and **PortNumber** is the port number used by the Oracle Application Server. The servlet name is case-sensitive, so be sure to use the correct case when typing the **mstrWebAdmin** or **MobileServlet name**.

2. When prompted for a user name and password, specify the values you used earlier when creating the user mapped to the admin security role (see **Deploying using the Oracle Enterprise Manager** above).
3 After you log in, the MicroStrategy Web Administrator page appears. Add and connect to an Intelligence Server.

If you are deploying MicroStrategy Mobile Server J2EE, after you are authenticated, the MicroStrategy Mobile Server Administrator page appears. Add and connect to an Intelligence Server. Once connected, click **Mobile Configuration** to configure your MicroStrategy Mobile applications to communicate with Mobile Server and Intelligence Server. For steps on how to define this configuration, see the *MicroStrategy Mobile Design and Administration Guide*. Creating a configuration completes the steps required to deploy Mobile Server J2EE.

4 You can now launch the MicroStrategy Web Universal project. For details, see *Launching the project* below.

### Launching the project

In a Web browser, access MicroStrategy Web Universal using this URL:

http://IPAddress:PortNumber/MicroStrategy/servlet/mstrWeb

where *IPAddress* is the IP address of the Oracle machine and *PortNumber* is the port number used by the Oracle Application Server.

The projects on the Intelligence Server appear, and you are ready to use MicroStrategy Web Universal.

### Deploying with JBoss (Windows)

This chapter provides information used to deploy and configure MicroStrategy Web Universal in a JBoss environment. You can also use the steps provided below to deploy MicroStrategy Mobile Server J2EE:

- **Preconfiguration information**: configuration that must occur before you begin deploying MicroStrategy Web Universal.

- **Deploying MicroStrategy Web Universal**: instructions for deploying, including step-by-step procedures.
Preconfiguration information

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Universal on JBoss on your machine.

This section uses the following configuration. While your setup may vary slightly, for example, you may have different versions of these applications, the overall process remains the same.

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Windows 2003 SP2 (on x86)</td>
</tr>
<tr>
<td>Application server</td>
<td>JBoss 4.2.3</td>
</tr>
<tr>
<td>JDK</td>
<td>SUN JDK 1.5.0 or 1.6</td>
</tr>
</tbody>
</table>

Configuring the JDK

If you have not installed SUN JDK 1.5 or 1.6 yet, download them from their respective websites:

- JDK 1.5.0: [http://java.sun.com/j2se/1.5.0/download.html](http://java.sun.com/j2se/1.5.0/download.html)

When you go to the download site, you may be presented with a number of software options. These might include terms such as JRE, JDK, and Java SDK. You must install a developer kit (which can be termed JDK or SDK) rather than installing only the JRE.

To configure the JDK, a system variable must point to the folder where you install the JDK. If you install the JDK to a simple folder path such as `C:\` then setting the system variable is easier and more likely to be correct.

After you install the Sun JDK, you must configure it.

To configure the JDK

1. On your Windows Desktop, right-click **My Computer** and select **Properties**. The System Properties dialog box opens.
2. Select the **Advanced** tab.
3 Click Environment Variables. The Environment Variables dialog box opens.

4 Under System Variables, click New to create a system variable. The New System Variable dialog box opens.

5 In the Variable Name box, type JAVA_HOME.

6 In the Variable Value box, specify the destination folder where you installed the JDK and click OK.

For example, if the fully qualified path to your JDK executable is C:\jsdk1.5.0\bin\java.exe, the value of your JAVA_HOME variable is C:\jsdk1.5.0.

If you have installed JDK under the Program Files folder, type Progra~1 in the destination folder; otherwise the system does not recognize the folder. For example, C:\Progra~1\jsdk1.5.0.

Installing JBoss

You can download and install JBoss from the following website:

http://sourceforge.net/project/showfiles.php?group_id=22866&package_id=16942&release_id=614346

Keep track of the location in which you install JBoss, as this location is used later (referred to as JBOSS_HOME) to configure JBoss with a MicroStrategy Web Universal deployment.

Locating the WAR file

The MicroStrategy Web Universal application is packaged within a single file, called a WAR file (Web ARchive), following the J2EE specifications, which you must deploy to run the application in your application server environment.

The MicroStrategy Web Universal WAR file (MicroStrategy.war) is located in the path you specified when installing MicroStrategy Web Universal. The default location when installing on 32-bit Windows environments is C:\Program Files\MicroStrategy\WebJSP. The default location when installing on 64-bit Windows environments is
C:\Program Files (x86)\MicroStrategy\WebJSP. The default location when installing on UNIX or Linux is described in Directory structure, page 127.

To deploy it, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this chapter.

**Locating the Mobile Server WAR file**

If you are deploying MicroStrategy Mobile Server J2EE, you must use the WAR file for Mobile Server J2EE. The WAR file is placed in the folder you specified when installing Mobile Server J2EE. You can then use the same processes used to deploy the MicroStrategy Web Universal WAR file to deploy the Mobile Server J2EE WAR file.

**Deploying MicroStrategy Web Universal**

Assuming you have made all the necessary configurations described above, you can begin deploying MicroStrategy Web Universal with JBoss. This involves the following steps:

1. **Deploying using JBoss as a stand-alone Web container**
2. **Controlling access to the MicroStrategy Web Administrator page**
3. **Accessing the MicroStrategy Web Administrator page**
4. **Launching the project**

**Deploying using JBoss as a stand-alone Web container**

To deploy MicroStrategy Web Universal using JBoss as a stand-alone Web container

1. Copy the MicroStrategy.war file to the JBOSS_HOME\server\default\deploy directory. Instructions on locating the MicroStrategy.war file can be found in the previous section, Locating the WAR file, page 282.
If you are deploying MicroStrategy Mobile Server J2EE instead, locate the `MicroStrategyMobile.war` file. Use this WAR file for the rest of the steps in this procedure.

2 To start JBoss, browse to `JBOSS_HOME\bin`. Then run the following command:

```
run.bat -b 0.0.0.0
```

MicroStrategy Web Universal is deployed automatically. There is now a `MicroStrategy` folder under `JBOSS_HOME\server\default\deploy` directory. If you deployed MicroStrategy Mobile Server J2EE instead, the folder is named `MicroStrategyMobile`.

**Controlling access to the MicroStrategy Web Administrator page**

The MicroStrategy Web Administrator page is accessible only to users with an “admin” role. To create a user name and password set that is authorized to access the Administrator page, you must create the users and assign them the role of admin under the JBoss user configuration files. The steps below show you how to give a user access to the MicroStrategy Web Administrator page. If you are deploying Mobile Server J2EE, you can use the steps below to provide access to the MicroStrategy Mobile Server Administrator page.

**To control access to the MicroStrategy Web Administrator page**

1 Browse to the directory `JBOSS_HOME\server\default\conf`, where `JBOSS_HOME` is the location in which you installed JBoss.

2 Create the following two files within this directory:
   - `users.properties`
   - `roles.properties`

3 Open the `users.properties` file in a text editor.

4 Include one line for each user to grant access to the Web Administrator page, using the following syntax:

```
user_id=user_password
```
For example, you create UserA and UserB with passwords 1234 and 5678 respectively using the following syntax:

```
UserA=1234
UserB=5678
```

5. Save your changes and close the `users.properties` file.

6. Open the `roles.properties` file in a text editor.

7. Include one line for each user you included in the `users.properties` file and grant them administrative access, using the following syntax:

```
user_id=admin
```

For example, you define UserA and UserB to have administrative access using the following syntax:

```
UserA=admin
UserB=admin
```

8. Save your changes and close the `roles.properties` file.

9. To start JBoss, browse to `JBOSS_HOME\bin`. Then run the following command:

```
run.bat -b 0.0.0.0
```

Now you can start MicroStrategy Web Administrator to add and connect to an Intelligence Server. To configure MicroStrategy Web Administrator, see *Accessing the MicroStrategy Web Administrator page, page 285.*

**Accessing the MicroStrategy Web Administrator page**

---

1. In a Web browser, access the administrator page by specifying the following URL:

- **For Web:** [http://localhost:8080/MicroStrategy/servlet/mstrWebAdmin](http://localhost:8080/MicroStrategy/servlet/mstrWebAdmin)
• For Mobile Server J2EE:
  

  The servlet names are case-sensitive. Make sure to use the correct case when typing the `mstrWebAdmin` or `MobileServlet` name. If the application server is enabled with security, a dialog box related to the administrator authentication opens.

  2 When prompted for a user name and password, type the user name for the administrator user you created in the `roles.properties` file and the login information in the `users.properties` file.

  3 Once you log in, the MicroStrategy Web Administrator page is displayed. Add and connect to an Intelligence Server.

  If you are deploying MicroStrategy Mobile Server J2EE, after you are authenticated, the MicroStrategy Mobile Server Administrator page appears. Add and connect to an Intelligence Server. Once connected, click **Mobile Configuration** to configure your MicroStrategy Mobile applications to communicate with Mobile Server and Intelligence Server. For steps on how to define this configuration, see the *MicroStrategy Mobile Design and Administration Guide*. Creating a configuration completes the steps required to deploy Mobile Server J2EE.

  4 Proceed to launch the MicroStrategy Web Universal project. For more information, see *Launching the project, page 286* below.

**Launching the project**

In a Web browser, access MicroStrategy Web Universal using this URL:


**Administering your MicroStrategy Web deployment**

Enabling users to install MicroStrategy Office from Web

From the MicroStrategy Web Administrator page, you can designate the installation directory path to MicroStrategy Office, and also determine whether or not a link to Office installation information appears in the MicroStrategy Web interface.

To specify the path to MicroStrategy Office and determine whether or not users can install MicroStrategy Office from Web

1. From the Windows Start menu, select Programs, MicroStrategy, Web, and then Web Administrator. The MicroStrategy Web Administrator page opens.
   
   If your server is not connected, click Connect.


3. In the Path to MicroStrategy Office Installation field, type the base URL of your MicroStrategy Web Services machine, for example:

   http://server:port/Web_Services_virtual_directory/Office

   MicroStrategy Web automatically attaches /Lang_xxxx/officeinstall.htm to the end of the URL, where Lang_xxxx refers to the currently defined language in MicroStrategy Web. For example, if the language in MicroStrategy Web is set to English, a completed URL may appear as follows:

   http://localhost/MicroStrategyWS/office/Lang_1033/officeinstall.htm

4. Test the URL path by clicking Go. If the path you specified is correct, the MicroStrategy Office Installation page is displayed.

5. Click your browser's Back button to return to the Web Administration - MicroStrategy Office settings page.
6 To ensure that an 'Install MicroStrategy Office' link is displayed at the top of users' project selection and login pages in MicroStrategy Web, select the **Show link to installation page for all users on the Projects and Login pages** check box. When users click the 'Install MicroStrategy Office' link, a page opens with instructions on how to install MicroStrategy Office on their machine.

7 Click **Save** to save the settings.

### Configuring your MicroStrategy installation

After completing the steps to deploy MicroStrategy Web or Web Universal, you can continue your setup and configuration. To help guide the rest of your installation and configuration steps, refer to the section **Installation and configuration checklists, page 87** in **Chapter 1, Planning Your Installation**, for installation and configuration checklists.
Introduction

This chapter describes the procedure to deploy MicroStrategy Web Services for ASP.NET and J2EE platforms. The process of deploying the ASP.NET version of MicroStrategy Web Services on Windows with Microsoft Internet Information Services (IIS) is explained in detail.

Steps to deploy MicroStrategy Web Services in a UNIX/Linux or Windows environment with selected Web and application servers is also explained in detail. MicroStrategy Web Services J2EE is platform-independent and can be deployed using different combinations of operating systems, Web servers, and application servers.

Web application servers are not MicroStrategy products, so detailed steps cannot be provided for every combination of application server and operating system. This chapter supplies instructions for a few of the most common combinations and procedures. The procedures for different operating systems and Web application server configurations are similar, but you should refer to the vendor-provided information and documentation for details specific to your configuration, or contact MicroStrategy Technical Support.
MicroStrategy Web Services provides a standard SOAP-based implementation of XML Web Services, which is used by MicroStrategy Office.

This chapter has the following sections:

- *Deploying with IIS (Windows), page 290*
- *General steps to deploy MicroStrategy Web Services J2EE, page 290*
- *Deploying with WebLogic, page 291*
- *Deploying with WebSphere and IBM HTTP Server, page 301*
- *Deploying with Sun Java System, page 311*
- *Deploying with Tomcat, page 320*
- *Deploying with SAP, page 324*
- *Deploying with Oracle 10g, page 327*
- *Completing your MicroStrategy Web Services J2EE deployment, page 331*
- *Administering your MicroStrategy Web Services deployment, page 333*
- *Configuring your MicroStrategy installation, page 334*

## Deploying with IIS (Windows)

MicroStrategy Web Services ASP.NET can be deployed with IIS only on Windows. This is handled automatically when you choose to install MicroStrategy Web Services ASP.NET during the installation process. See *Chapter 2, Installing MicroStrategy on Windows* for details on installing MicroStrategy on Windows.

## General steps to deploy MicroStrategy Web Services J2EE

After you have installed MicroStrategy Web Services J2EE, you can deploy and configure it for your specific environment. The configuration and
deployment steps are provided in the Web server and application server sections in this chapter. The table below lists the general steps for all environments.

<table>
<thead>
<tr>
<th>High-Level Deployment Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Log on to the application server by using the proper user name and password.</td>
</tr>
<tr>
<td>3 To increase the performance of the application before proceeding with the deployment, see the Performance-based setup information section, if available, for your environment and configure as necessary. Also, after deploying MicroStrategy Web Services J2EE on your machine, you may need to complete a few performance-based setup steps.</td>
</tr>
<tr>
<td>4 Choose the desired deployment method.</td>
</tr>
<tr>
<td>5 Follow the deployment procedure.</td>
</tr>
<tr>
<td>6 View the MicroStrategy Web Services home page and validate your installation.</td>
</tr>
<tr>
<td>7 Complete your installation by validating access to the runtime files, editing your project sources file, and testing your Web Services installation.</td>
</tr>
</tbody>
</table>

## Deploying with WebLogic

This section provides the information used to deploy and configure MicroStrategy Web Services J2EE using Oracle WebLogic as the application server. It provides the instructions for WebLogic 10.3 and assumes that your application server is already installed and operational.

For a list of application servers that are certified to work with MicroStrategy Web Services J2EE, see *MicroStrategy Web Services J2EE software requirements and recommendations, page 64.*

This section includes the following information:

- **WebLogic paths and folder locations, page 292:** Default folder structure for WebLogic.
- **Preconfiguration information, page 293:** Configuration that must occur before you begin deploying MicroStrategy Web Services J2EE.
- **Deploying MicroStrategy Web Services J2EE, page 294:** Instructions for deploying the application.
- **Redeploying the application, page 300:** Instructions for re-deploying the application.
• **Performance-based setup information, page 300**: Optional configuration settings to increase the application’s performance.

The additional configuration steps are not required to run MicroStrategy Web Services J2EE, but these settings can improve its performance. Review the performance-based setup information prior to deploying the system to see if these changes are of interest to you.

# WebLogic paths and folder locations

This section presents the default folder structure for WebLogic and provides the variables used throughout the chapter to represent the WebLogic mydomain folder path.

Each version of WebLogic is installed with a different default path to the WebLogic mydomain folder. When deploying MicroStrategy Web Services J2EE, you must make some changes within the WebLogic folders. Thus, it is important to understand the WebLogic folder structure for the version of WebLogic you are using. The default folder structure for WebLogic 10.3 is: WEBLOGIC_HOME\weblogic10\wlserver_10.3\common\bin

Note the following:

- WEBLOGIC_HOME is the WebLogic Server home path.

- The folder structures are configurable and your organization may have changed the default names or path.

If you cannot view the user_projects folder after installing the application, perform the steps listed below.

To create the folder structure using the Oracle WebLogic Configuration Wizard

These steps may not work for your version and configuration of WebLogic. If the following steps do not work, refer to your vendor documentation.

1. Start the WebLogic configuration by executing the config.cmd from WEBLOGIC_HOME\weblogicversion\common\bin folder. The Oracle Weblogic Configuration Wizard opens to the Create or Extend a Configuration page.
Select the **Create a new Weblogic configuration** option and click **Next**. The Select a Configuration Template page opens.

Ensure to select the **Generate a domain configured automatically to support the following Oracle products** option. Click **Next**.

Enter a user name and password, and click **Next**. The Configure Server Start Mode and Java SDK page opens.

Select the **Development** or **Production** mode option as appropriate.

Select the Java SDK version and click **Next**. The Create Weblogic Configuration page opens.

When prompted with the option **Do you want to customize any of the following options?**, select **No**. Click **Next**.

Enter name and location for the **Domain Name** and **Domain Location** fields and then click **Create**.

Throughout this chapter, the WebLogic mydomain folder is referred to as **WEBLOGIC_MYDOMAIN_FOLDER**.

**Preconfiguration information**

This section provides the preconfiguration information to deploy MicroStrategy Web Services J2EE on your machine. This includes *Locating the WAR file, page 294*.

Note the following:

- For information on the version numbers of application servers, operating systems, and JVMs supported or certified by MicroStrategy, see the MicroStrategy readme.

- For information on installing WebLogic products, see http://www.oracle.com/technology/products/weblogic/integration/index.html

- Before you start the deployment process, locate the machine name and IP address.
Locating the WAR file

The MicroStrategy Web Services J2EE application is packaged within a single file, called a WAR (Web ARchive) file, following J2EE specifications. You must deploy the WAR file to run the application in your application server environment. For more information on installation on UNIX or Linux, see *Installation procedures on UNIX and Linux, page 112*. For more information on installation on Windows, see *Installation procedure, page 92*.

The WAR file is placed in the folder you specified when installing MicroStrategy Web Services J2EE. Copy the WAR file to the `WEBLOGIC_MYDOMAIN_FOLDER`. See *WebLogic paths and folder locations, page 292* for information on the default folder structure.

To deploy the WAR file, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You can also refer to the section *Deploying MicroStrategy Web Services J2EE, page 294* for more details on deploying the WAR file.

After deploying the `MicroStrategyWS.war` file, you can view the `WEB-INF` folder, which contains the subfolders `classes` and `lib`.

Deploying MicroStrategy Web Services J2EE

You can deploy MicroStrategy Web Services J2EE using one of the following deployment methods:

- The automatic deployment (development mode) feature is the easiest and fastest way. To deploy in development mode, you can set the `PRODUCTION_MODE` parameter value as false or blank in the `startWebLogic` file in `WEBLOGIC_MYDOMAIN_FOLDER/` (for UNIX/Linux the file is `startWebLogic.sh`, for Windows the file is `startWebLogic.cmd`). For more information, see *Deploying automatically (development mode), page 296*.

- The manual deployment feature can be used for environments where the server is running in production mode, and automatic deployment is turned off. To deploy in production mode, you can set the `PRODUCTION_MODE` parameter value as true in the `startWebLogic` file in `WEBLOGIC_MYDOMAIN_FOLDER/` (for UNIX/Linux the file is...
When your machine has been configured with the necessary settings, you can deploy MicroStrategy Web Services J2EE with WebLogic. This involves the following steps:

1. **Starting the WebLogic application server, page 295**

2. **Deploying automatically (development mode), page 296**
   - or -
   **Deploying manually (production mode), page 297**

3. **Viewing the MicroStrategy Web Services home page, page 299**

4. **Redeploying the application, page 300**

Note the following:

- The *Performance-based setup information, page 300* section provides information on additional settings to improve application performance. These additional settings are not required but can improve the performance of MicroStrategy Web Services J2EE. Review this information prior to deployment to see if these options are of interest to you.

**Starting the WebLogic application server**

After choosing the deployment mode as development or production, you must start the WebLogic application server. You can start the WebLogic application server by using the `startWebLogic` file in `WEBLOGIC_MYDOMAIN_FOLDER/` (for UNIX/Linux the file is `startWebLogic.sh`, for Windows the file is `startWebLogic.cmd`).
Deploying automatically (development mode)

When the deployment is set to development mode, the application is automatically deployed, as soon as you place the WAR file or the exploded directory in the WEBLOGIC_MYDOMAIN_FOLDER/autodeploy folder.

With this method, you can deploy from an exploded directory, where all the files contained in the WAR file are extracted. When you deploy from an exploded directory, all of the files and folders within the WAR file are exposed to WebLogic.

For WebLogic 9.x and later, copy the exploded directory extracted from MicroStrategyWS.war file to the WEBLOGIC_MYDOMAIN_FOLDER/autodeploy folder.

To deploy MicroStrategy Web Services J2EE from the exploded directory

The WAR file must be uncompressed by the same user who started the application.

1. Locate the MicroStrategyWS.war file. It is located in the MicroStrategy Web Services J2EE deployment directory you specified during installation. For more information on locating the WAR file, see Locating the WAR file, page 294.

2. Create the following new folder:

   /home/username/context_folder

   where username is your account name used to access the Web Services server machine, and context_folder is the name of the new folder.

   You can create the new folder anywhere except in the following location:

   /WEBLOGIC_MYDOMAIN_FOLDER/autodeploy

3. Copy the WAR file to the new folder.

4. To explode the WAR file in the folder you created:

   • For UNIX/Linux, run the following command from the command line:

     jar -xvf MicroStrategyWS.war
• For Windows, rename the MicroStrategyWS.war file to MicroStrategyWS.zip before unzipping the file.

5 Delete the MicroStrategyWS.war file from the new folder, if it exists in the folder.

6 Move the new folder to the WEBLOGIC_MYDOMAIN_FOLDER/autodeploy folder.

The application is automatically deployed. To view your Web Services home page, see Viewing the MicroStrategy Web Services home page, page 299.

To enhance the performance of MicroStrategy Web Services J2EE, you can configure additional settings after deployment. For more information, see Performance-based setup information, page 300.

Deploying manually (production mode)

With manual deployment, you can deploy MicroStrategy Web Services J2EE from an exploded directory, where all the files contained in the WAR file are extracted. When you deploy from an exploded directory, all of the files and folders within the WAR file are exposed to WebLogic. This allows WebLogic to access the required folders to perform any necessary configurations to files in the exploded directory.

Perform the deployment in the WEBLOGIC_MYDOMAIN_FOLDER/applications directory.

To manually deploy MicroStrategy Web Services J2EE from the exploded directory

1 Locate the MicroStrategyWS.war file. It is located in the MicroStrategy Web Services J2EE deployment directory you specified during installation.

2 Create a new folder in the WEBLOGIC_MYDOMAIN_FOLDER/autodeploy directory and transfer the WAR file to this directory.

3 Unzip the WAR file using the following command from the command line:

   jar -xvf MicroStrategyWS.war
4  Delete the MicroStrategyWS.war file from the new folder.

5  Open the WebLogic Server (WLS) Admin Console by accessing the following address:

   http://IP address:port/console/

   where IP address is the IP address of the machine on which you installed the WebLogic application server and port is the port number for the WebLogic application server.

6  Type a valid user ID and password at the prompt. The user ID and password are the ones you specified when installing the WebLogic application server on your machine.

7  To complete this operation, see Configure from the WebLogic Server Admin Console, page 298.

**Configure from the WebLogic Server Admin Console**

To configure from the WebLogic Server (WLS) Admin Console, refer to the procedure below.

---

**To configure from the WLS Admin Console with WebLogic 8.x**

1  Open the WLS Admin Console by accessing the following address:

   http://IP address:port/console/

   where IP address is the IP address of the machine on which you installed the WebLogic application server and port is the port number for the WebLogic application server.

2  Select mydomain, point to Deployments, and then choose Web Application Modules. The Web Application table opens.

3  Select Deploy a New Web Application Module.

4  Locate and select the WAR file to configure and deploy. If you are deploying from the exploded directory, select the root of the exploded directory.

5  Click the Target Module button.
6 Click **Deploy**, then wait for the application to activate.

To enhance the performance of MicroStrategy Web Services J2EE, you can configure additional settings after deployment. For more information, see *Performance-based setup information, page 300.*

**Viewing the MicroStrategy Web Services home page**

Viewing the home page offers a first level of validation of MicroStrategy Web Services J2EE.

**To view the MicroStrategy Web Services home page**

1 The address to launch MicroStrategy Web Services home page is different for each deployment method. The following table lists the various URLs to access MicroStrategy Web Services using a WAR file or an exploded directory.

Names are case-sensitive. Make sure to use the correct case when typing. If the application server is enabled with security, a dialog box related to the administrator authentication opens.

<table>
<thead>
<tr>
<th>Deployment Method</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>When deployed from a WAR file</td>
<td>Access the home page from a browser using either of the following addresses:</td>
</tr>
<tr>
<td></td>
<td>• <a href="http://IPaddress:port/context_name/">http://IPaddress:port/context_name/</a></td>
</tr>
<tr>
<td></td>
<td>• <a href="http://IPaddress/context_name/">http://IPaddress/context_name/</a></td>
</tr>
<tr>
<td></td>
<td>The first URL accesses the Web Services home page through port for the</td>
</tr>
<tr>
<td></td>
<td>WebLogic application server. The second URL uses the Apache Web server</td>
</tr>
<tr>
<td></td>
<td>when WebLogic is configured with Apache Web server.</td>
</tr>
<tr>
<td></td>
<td>The context_name is the name you gave to the WAR file and IPaddress is</td>
</tr>
<tr>
<td></td>
<td>the IP address of your machine. By default, the context_name is</td>
</tr>
<tr>
<td></td>
<td>MicroStrategyWS.</td>
</tr>
<tr>
<td>When deployed from an exploded</td>
<td>Access the home page from a browser at this address:</td>
</tr>
<tr>
<td>directory</td>
<td><a href="http://IPaddress:port/context_folder/">http://IPaddress:port/context_folder/</a></td>
</tr>
<tr>
<td></td>
<td>where context_folder is the name of the folder where the application was</td>
</tr>
<tr>
<td></td>
<td>exploded and IPaddress is the IP address of your machine.</td>
</tr>
</tbody>
</table>

You can now view the MicroStrategy Web Services home page. You have completed the steps required to deploy MicroStrategy Web Services J2EE. To validate access to the required runtime files, edit your project sources file
and test your MicroStrategy Web Services J2EE installation, refer to the section *Completing your MicroStrategy Web Services J2EE deployment*, page 331.

## Redeploying the application

If you have already deployed MicroStrategy Web Services J2EE with WebLogic and you change any parameters in the `web.xml` file, you must re-deploy the application using the WLS Admin Console. This allows the changes to take effect in the deployed application.

### To redeploy the application from the WLS Admin console

1. Open the WLS Admin Console by accessing the following address:

   ```
   http://IP address:port/console/
   ```

   where **IP address** is the IP address of the machine on which you installed the WebLogic application server and **port** is the port number for the WebLogic application server.

2. Type a valid user ID and password at the prompt. The user ID and password refer to the details you specified when installing the WebLogic application server on your machine.

3. In the left frame, select **mydomain**, point to **Deployments**, and then choose **Web Application Modules**. The Web Application table opens, which lists the MicroStrategy Web Services J2EE applications that have been deployed.

4. Click the application that you want to re-deploy, such as **MicroStrategyWS**. The information for the application appears.

5. In the right frame, select the **Deploy** tab.

6. Click **Redeploy**. The application is redeployed.

## Performance-based setup information

The performance of MicroStrategy Web Services J2EE can be enhanced in certain cases with configurations to various components. These additional
configurations are not required. This section provides the following configurations intended to enhance the performance of Web Services J2EE:

- **Setting the Java heap size**

### Setting the Java heap size

The Java heap size for the WebLogic Server can be increased by modifying the `JAVA_OPTIONS` variable in the `startWebLogic` file.

#### To increase the Java heap size

1. Open the `startWebLogic` file in the folder 
   `WEBLOGIC_MYDOMAIN_FOLDER/`.
   For UNIX/Linux the file is `startWebLogic.sh`, for Windows the file is `startWebLogic.cmd`.

2. Look for the following line in the file:
   
   ```
   set JAVA_OPTIONS = "-ms128m -mx128m"
   ```

   This line reflects a Java heap size of 128 MB. Set the appropriate size. MicroStrategy recommends the initial Java heap size to be set at a minimum of 500MB, assuming the machine has enough memory space.

   If the `JAVA_OPTIONS` parameter is not set, you can add the line specified in step 2, for setting the options.

3. Stop and start the application server.

### Deploying with WebSphere and IBM HTTP Server

This section provides the information used to deploy and configure MicroStrategy Web Services J2EE using the WebSphere 6.x Server and the IBM HTTP Web Server.
For a list of application servers that are certified to work with MicroStrategy Web Services J2EE, see MicroStrategy Web Services J2EE software requirements and recommendations, page 64.

This section includes the following information:

- **Preconfiguration information, page 302**: Configuration that must occur before you begin deploying MicroStrategy Web Services J2EE.
- **Deploying MicroStrategy Web Services J2EE, page 303**: Instructions for deploying the application.
- **Performance-based setup information, page 310**: Optional settings to increase the application’s performance.

These additional settings are not required, but can enhance the performance of MicroStrategy Web Services J2EE. Review this information prior to deployment to see if any of these options are of interest to you.

## Preconfiguration information

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Services J2EE on your machine. Preconfiguration includes *Locating the WAR file, page 302*.

This section also provides additional setup information for the machine where the application server and Web server are already installed.

## Locating the WAR file

The MicroStrategy Web Services J2EE application is packaged within a single file, called a WAR (Web ARchive) file, following the J2EE specifications. You must deploy this file to run the application in your application server environment.

### To locate the WAR file

1. Locate the MicroStrategyWS.war file in the MicroStrategy Web Services J2EE deployment directory you specified during installation. For more information on installation on UNIX or Linux, see *Installation*.
procedures on UNIX and Linux, page 112. For more information on installation on Windows, see Installation procedure, page 92.

2 Copy the WAR file to the \texttt{WAS\_HOME/installableApps} directory, where \texttt{WAS\_HOME} is the home path of the WebSphere 6.x application server.

The deployment process in this chapter assumes that the MicroStrategyWS.war file is stored in this directory.

To deploy the WAR file, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this chapter.

After deploying MicroStrategyWS.war file, you can view the WEB-INF folder, which contains subfolders classes and lib. Logs are created and maintained in the application server root folder.

### Deploying MicroStrategy Web Services J2EE

Once your machine has the necessary settings configured, you can deploy MicroStrategy Web Services J2EE. Deployment involves the following steps:

1 Launching the WebSphere Administrative Console, page 304
2 Starting the WebSphere default application server, page 304
3 Installing the Enterprise Application, page 305
4 Preparing for the application installation, page 305
5 Restarting the application server, page 307
6 Starting the Web module, page 308
7 Viewing the MicroStrategy Web Services home page, page 309

The Performance-based setup information, page 310 section provides information on additional settings to increase the application performance. These additional settings are not required, but can increase the performance of MicroStrategy Web Services.
J2EE. Review this information prior to deployment to see if any of these options are of interest to you.

### Launching the WebSphere Administrative Console

The WebSphere Administrative Console can be accessed only if the WebSphere server is started on the machine.

---

**To launch the WebSphere Administrative Console**

In a Web browser, type the URL for the administrative console. For HTTP, the default port in WebSphere 6.x is 9060. The URL is of the following form:

http://HostName:9060/admin

Enter the value for the user ID parameter. For example, admin.

### Starting the WebSphere default application server

After you launch the WebSphere Administrative Console, you can deploy MicroStrategy Web Services J2EE by starting the default application server.

---

**To start the default application server**

When the WebSphere Administrative Console opens, a tree view is displayed in the left frame.

1. Expand the **Servers** node, or click the link to expand the view.

2. Click the **Applications Servers** link. A table listing the application servers is displayed to the right of the navigation tree. This area is the Workspace.

3. Select the check box next to the application server to be started.

4. Click **Start** above the table.
If your version of WebSphere does not offer the start and stop options through the console, use the following command line options:

- For Windows:
  - `startServer servername` to start the application server.
  - `stopServer servername` to stop the application server.

- For UNIX/Linux:
  - `startServer.sh servername` to start the application server.
  - `stopServer.sh servername` to stop the application server.

where `servername` is the name of the default application server.

**Installing the Enterprise Application**

**To install the Enterprise Application**

1. Expand **Applications**, and then **Enterprise Applications** to display a list of installed applications.

2. Click **Install**.

**Preparing for the application installation**

The following steps describe the settings that must be specified for the installation.

This procedure provides general requirements to prepare for the installation of MicroStrategy Web Services J2EE with WebSphere. Additional configuration may be required depending on the requirements of your production environment.

**To specify settings for the installation**

1. Specify the path to the WAR file by selecting the **Specify path** option, from either the **Local file system** option or the **Remote file system** option.
2 Type in the full path for the location of the `MicroStrategyWS.war` file.

The URL to access MicroStrategy Web Services J2EE 
(http://machine-name/ContextRoot/) contains 
ContextRoot, which should be replaced by any name of your 
choice. ContextRoot is case-sensitive. For example, you can use 
the default name of the WAR file, which is `MicroStrategyWS`.

3 Click **Next**.

4 Select **Do not override existing bindings**.

5 Click the radio button **Use default virtual host name for web modules**.

6 Click **Next**. An installation options screen opens.

   If the installation options screen is not displayed, click **Continue** 
on the Application Security Warnings screen to view the 
installation options screen.

7 Make the following changes:

   • Specify the value for the **Directory to install application** as 
     ${APP_INSTALL_ROOT}/DefaultNode

   • Specify an **Application name** of your choice.

   • Enter the value of -1 as the **Reload interval in seconds**.

8 Click **Next**. The Map modules to servers page opens.

9 Select the check box for MicroStrategy Web Services J2EE module, select 
the application server, and then click **Apply**.

10 Click **Next**. The Map virtual hosts for Web modules page opens.

11 In the right navigation pane, expand the **Apply Multiple Mappings** node. 
Map the admin role to the users or groups who are given the 
MicroStrategy Web Services Administrator privileges. Role mapping 
enables the mapped users to access the MicroStrategy Web Services 
home page.

   Security must be enabled on the WebSphere Server for this feature 
to work.

12 Select the check box for MicroStrategy Web Services J2EE web module 
and then click **Next**. The Summary Page opens.
13 Review the summary and click **Finish**.

14 A message appears stating that the installation was successful. Click **Save to Master Configuration** to save the changes to the master repository.

15 Click the **Save** button that appears.

**To configure class loading**

16 Expand **Applications**, and then **Enterprise Applications** to display a list of installed applications.

17 In the list of Enterprise Applications, click on the application name just installed, for example, **MicroStrategyWS_war**.

18 In the **Detail Properties** list, click **Class loading and update detection**.

19 For the **Class loader order**, select **Classes loaded with application class loader first**.

20 Click **Apply**. The page on the right pane returns to its previous main configuration tab.

21 In **Modules**, click **Manage Modules**.

22 Under the **Module** header, click the **MicroStrategy Web Services J2EE** hyperlink.

23 In the **Class loader order** drop-down list, select **Classes loaded with application class loader first**.

24 Click **Apply**.

25 In the **Message** box at the top of the page, click **Save**.

26 Click the **Save** button that appears.

**Restarting the application server**

This section explains how to stop and start the application server. Performing these steps stops and starts all the applications running on the application server. To stop and start only the application in which you are working, see *To start the Web module, page 308*. 

© 2011 MicroStrategy, Inc. Deploying with WebSphere and IBM HTTP Server 307
To restart the application server

Some versions of WebSphere provide the means to start and stop the server from within the console.

1. Expand **Servers**, and then **Application servers**. A table listing the application servers and an icon indicating their status appears:
   - Red: stopped
   - Green: started

2. Select the check box next to the application server to stop, and click **Stop**. The status icon changes from green to red.

3. Select the check box next to the application server to start and click **Start**. The application server starts and the status icon changes to green.

If your version of WebSphere does not offer the start and stop options through the console, use the following command line options:

   - For Windows:
     - `startServer servername` to start the application server.
     - `stopServer servername` to stop the application server.

   - For UNIX/Linux:
     - `startServer.sh servername` to start the application server.
     - `stopServer.sh servername` to stop the application server.

   where `servername` is the name of the application server.

Starting the Web module

This process starts only the Web application, rather than all the applications running on the application server. To stop and start all applications, see **Restarting the application server** above.

To start the Web module

1. Expand **Applications**.
2 Click the **Enterprise Applications** link. A list of installed applications are displayed in the table to the right, along with icons indicating their status:

- Red: stopped
- Green: started

3 Select the check box next to the application to start, and click **Start**.

**Viewing the MicroStrategy Web Services home page**

Viewing the MicroStrategy Web Services J2EE home page is the first level validation of your installation. The home page is accessible only to users with an **admin** role. To create the set of users and passwords that are authorized to access the administrator page, you must create the necessary role mapping between these users to the **admin** role for the MicroStrategy Web Services J2EE application. The steps to perform this setup are given above in the section *Preparing for the application installation, page 305*. For more information, you can refer to your IBM documentation or contact MicroStrategy Technical Support.

---

**To access the MicroStrategy Web Services home page**

1 Access the home page by typing the following URL in a web browser:

```
http://WebServerName:Port/ContextRoot/
```

where **port** is the HTTP transport port of your application (default port is 9080), and **ContextRoot** is the name you provided for the **ContextRoot** for Web Module box in the section *Preparing for the application installation, page 305*. For example, the default name of the WAR file, which is **MicroStrategyWS**.

- The servlet and JSP names are case-sensitive. If the application server is enabled with security, a dialog box related to the administrator authentication opens.

2 Type the user ID and password assigned with an **admin** role.

After you are authenticated, the MicroStrategy Web Services J2EE home page opens. Use it to explore Web Services, validate your installation, and test Web Services.
You have completed the steps required to deploy MicroStrategy Web Services J2EE. To validate access to the required runtime files, edit your project sources file, and test your MicroStrategy Web Services J2EE installation, refer to the section *Completing your MicroStrategy Web Services J2EE deployment, page 331.*

Uninstalling the MicroStrategy Web Services J2EE application

You can uninstall the MicroStrategy Web Services J2EE application through the WebSphere Administrative console.

To uninstall the MicroStrategy Web application

1. Expand **Applications**.
2. Click **Enterprise Applications**. A list of installed applications are displayed in the table to the right.
3. Select the check box of the desired MicroStrategy Web Services J2EE application.
4. Click **Uninstall**.
5. Save the configuration in the master repository.

Performance-based setup information

The performance of MicroStrategy Web Services J2EE can be enhanced in certain cases with configurations to various components. These additional configurations are not required. This section provides the following configurations intended to enhance the performance of Web Services J2EE:

- Setting the Java heap size, page 310

Setting the Java heap size

You can increase the Java heap size for a given application server by configuring the WebSphere Administrative Console.
To increase the Java heap size

1. Access the Administrative Console.
2. Expand the **Servers** node.
3. Click the **Application Servers** link to view the list of application servers.
4. Click the application server name, and scroll to **Server Infrastructure**.
5. Expand **Java and Process Management**, and click **Process Definition**.
6. Click **Java Virtual Machine** to set the Java heap size settings.
   MicroStrategy recommends that you initially set the java heap size to a minimum of 500MB, assuming the machine has enough memory space.
7. Click **Apply** and save your changes.
8. Stop and start the application server.

Deploying with Sun Java System

This section provides the information used to deploy and configure MicroStrategy Web Services J2EE on a Sun Java System Application Server. It includes the following information:

- **Preconfiguration information, page 312**: Configuration that must occur before you begin deploying MicroStrategy Web Services J2EE.
- **Deploying MicroStrategy Web Services J2EE, page 313**: Instructions for deploying.
- **Performance-based setup information, page 319**: Optional settings to increase the application’s performance.

   These additional settings are not required but can enhance the performance of MicroStrategy Web Services J2EE. Review this information prior to deployment to see if any of these options are of interest to you.
- **Undeploying MicroStrategy Web Services J2EE using the command line utility, page 319**: Instructions for undeploying MicroStrategy Web Services J2EE.
Preconfiguration information

This section provides the preconfiguration information necessary for deploying MicroStrategy Web Services J2EE on your machine. This includes:

- Locating the WAR file, page 312
- Updating the server.policy file to support Web Services on Sun Java System Application Server 9.x, page 313

It also provides additional setup information that needs to be done on the machine on which the application server and web server are already installed.

Locating the WAR file

The MicroStrategy Web Services J2EE application is packaged within a single file, called a WAR file (Web ARchive), following the J2EE specifications. You must deploy this WAR file to run the application in your application server environment.

The MicroStrategy Web Services J2EE WAR file (MicroStrategyWS.war) is located in the MicroStrategy Web Services J2EE deployment directory you specified during installation. For more information on installation on UNIX or Linux, see Installation procedures on UNIX and Linux, page 112. For more information on installation on Windows, see Installation procedure, page 92.

To deploy the WAR file, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this chapter.

After deploying the MicroStrategyWS.war file, you can view the WEB-INF folder. This folder contains subfolders classes and lib. Log files are created and maintained in the application server root folder.
Updating the server.policy file to support Web Services on Sun Java System Application Server 9.x

To successfully deploy MicroStrategy Web Services J2EE on Sun Java System Application Server 9.x, you must modify the Sun Java System Application Server server.policy file. The file is located in the following path within the deployment directory for the Sun Java System Application Server:

..\AppServer\domains\domain1\config\server.policy

Add the following code to the end of the server.policy file.

```java
//Fix mstrws axis logger problem
grant {
permission java.lang.RuntimePermission "createClassLoader";
permission java.lang.RuntimePermission "getClassLoader";
permission java.util.PropertyPermission "*", "read,write";
};
```

Deploying MicroStrategy Web Services J2EE

Once your machine has been configured, you can deploy MicroStrategy Web Services J2EE with Sun Java System Application Server 9.x.

The deployment involves the steps below, which are explained in detail in the following sections:

1. *Launching the Sun Java System Application Server Admin Console*, page 314
3. *Controlling access to the MicroStrategy Services home page*, page 315
4. *Viewing the MicroStrategy Web Services home page*, page 317
5. *Restarting the application server*, page 318

The administration and deployment tools in Sun Java System Application Server have the same interface regardless of the
UNIX/Linux operating system on which they run. Therefore, the deployment process is the same for all UNIX/Linux operating systems, and is described below. The process on a Windows operating system has some minor differences, which are highlighted where necessary.

**Launching the Sun Java System Application Server Admin Console**

This procedure describes the steps to launch the Sun Java System Application Server.

---

**To launch the Sun Java System Application Server Admin Console**

1. Start the Sun Java System Administration Server. To start the server, navigate to the following directory from the command prompt:

   \`InstallDir/bin\`

   and type the following command:

   \`asadmin start-domain domain1\`

   where \`domain1\` refers to the default domain server or the server you are currently working on.

   To start the server in a Windows environment, from the Start menu, point to **Programs, Sun Microsystems, Application Server 9.x**, and then choose **Start Default Server**.

2. Access the Sun Java System Application Server Admin Console by typing the following URL:

   \`http://machine_name:admin_port\`

   where \`machine_name\` is the IP address or the name of the machine where you installed Sun Java System Application Server, and \`admin_port\` is the administration server port you provided during the installation. By default, the admin port is 4848.

   To access the Application Server Admin Console in a Windows environment, from the Start menu, point to **Programs, Sun Microsystems, Application Server 9.x**, and then choose **Admin Console**.
3 Enter the user name and password that you provided during the installation.

By default, the user name is admin. There is no password set by default. It is mandatory for you to assign a password during the installation.

Deploying MicroStrategy Web Services J2EE

After launching the Sun Java System Application Server Admin Console, follow the steps below to deploy MicroStrategy Web Services J2EE a WAR file.

To deploy MicroStrategy Web Services J2EE as a WAR file

1 Access the Admin Console.

2 Enter the user name and password that you assigned during the installation. By default, the user name is admin.

3 Expand Applications.

4 Select Web Applications and click Deploy. The Deploy Web Module page opens in the right pane.

5 Type the full path or browse to the location where you want to deploy the WAR file.

6 Click Next.

7 For the context and application name, use MicroStrategyWS, which is the default.

8 All the other fields are optional and can be set as per your requirements.

9 Click OK.

Controlling access to the MicroStrategy Services home page

For security purposes, you should only assign certain users the authorization to access the MicroStrategy Web Services J2EE home page. To do this, map
the Admin security role to only those users for whom you want to grant permission to work in the MicroStrategy Web Services home page.

Sun Java System Application Server supports the following authentication realms:

- File
- Admin-realm
- Certificate

A realm, also called a security policy domain or security domain, is a scope over which a common security policy is defined and enforced by the security administrator of the security service. For more information, refer to your application server vendor documentation. For details, you can also refer to the following resource:

http://docs.sun.com

In Sun Java System Application Server, the file realm is selected by default. To create users in the default realm, click File, then click Manage Users to start creating users authorized to access the MicroStrategy Web Administrator page.

**Mapping a security role**

A descriptor file, sun-web.xml, enables Sun Java System Application Server to map the existing users or groups to different security roles. This file is located in the WebApplicationRootDir/WEB-INF folder. You can modify this file, if you want to change the default values for the security roles.

The contents of this file are as follows:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sun-web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Application Server 8.1 Servlet 2.4//EN" "http://www.sun.com/software/appserver/dtlds/sun-web-app_2_4-1.dtd">

<sun-web-app xmlns="http://java.sun.com/xml/ns/j2ee">
  <context-root>/MicroStrategyWS</context-root>
  <security-role-mapping>
```

For mapping security roles to users and user groups, the `sun-web.xml` file assumes the default values are provided during installation of Sun Java System Application Server 9.x and deployment of MicroStrategy Web Services J2EE. The following are the values taken for context-root, role mapping, and the user or principal user:

- **Context-root** is set to `MicroStrategyWS`. This is the default value for the context name when deploying MicroStrategy Web Services J2EE.
- **Role mapping** is set to `admin`. This is the default value specified in the `web.xml` file.
- **The user or principal name** is also set to `admin`. This is the default name assigned to the user when installing Sun Java System Application Server 9.x.

### Viewing the MicroStrategy Web Services home page

Make sure your user has the proper privileges to access the MicroStrategy Web Services J2EE home page.

#### To access the MicroStrategy Web Services home page

1. Access the home page by typing the following URL in a Web browser:

   ```
   http://machine:port/ContextRoot/
   ```

   where `port` is the HTTP/HTTPS transport port of your application server (default port is 8080 for HTTP and 8181 for HTTPS), and `ContextRoot` is the name provided for the `ContextRoot` during installation. By default, this name is `MicroStrategyWS`.  

```xml
<role-name>admin</role-name>

<principal-name>admin</principal-name>

</security-role-mapping>

</sun-web-app>
```
You have completed the steps required to deploy MicroStrategy Web Services J2EE. To validate access to the required runtime files, edit your project sources file and test your MicroStrategy Web Services J2EE installation, refer to the section *Completing your MicroStrategy Web Services J2EE deployment, page 331*.

**Restarting the application server**

You must stop and restart the application server instance to apply the changes. This is necessary after making most of the changes to the application server configuration, such as creating a user. If the server must be restarted, the **Restart Required** message is displayed.

---

**To restart the application server**

You can restart the application server by navigating through the menu on the left, as follows:

1. Expand **Applications**.

2. Select the server instance you are currently working on and click the **Stop Instance** button displayed on the right panel. The Sun Java System Application Server Administration Console window opens.

3. Follow the instructions given in the Sun Java System Application Server Administration Console window to stop and restart the application server.

After restarting the application server, refer to the section *Completing your MicroStrategy Web Services J2EE deployment, page 331* to connect to the project page.

You are now ready to use MicroStrategy Web Services J2EE.
Performance-based setup information

The performance of MicroStrategy Web Services J2EE can be enhanced in certain cases with configurations to various components. These additional configurations are not required. This section provides the following configurations intended to enhance the performance of Web Services J2EE:

- Setting the Java heap size, page 319
- Undeploying MicroStrategy Web Services J2EE prior to upgrading, page 319

Setting the Java heap size

The maximum Java heap size, which handles all the Java requests, is set to 512MB by default.

To set the Java heap size

1) Navigate to Application server.

2) Click JVM Settings and select JVM Options. You can view the argument -Xmx512. This is the default value for the maximum heap size.

3) Click Add JVM Option to include additional arguments using -Xmx and -Xms. For example, if you want to set the maximum heap size to 1GB and minimum heap size to 512MB, then the argument you provide is -Xmx1024m and -Xms512m.

Undeploying MicroStrategy Web Services J2EE prior to upgrading

Oracle recommends undeploying an application before deploying a newer version.

Undeploying MicroStrategy Web Services J2EE using the command line utility

To undeploy an existing MicroStrategy Web Services J2EE application, use the following command:
asadmin undeploy --user admin-user --password password --port admin_port MicroStrategyWS

where admin-user, password, and admin_port are the user name, password, and admin port number that you provided during the installation. By default, the admin port is 4848. You are not required to provide the path to the WAR file.

After the undeployment is finished, stop and restart the application server for the changes to take effect.

**Undeploying MicroStrategy Web Services J2EE using the Admin console**

You can also undeploy the application by navigating through the Admin console menu on the left.

---

**To undeploy MicroStrategy Web Services J2EE using the Admin console**

1. Expand Applications.

2. Click Web Applications. Select the check box for MicroStrategyWS and click Undeploy.

3. After the undeployment is finished, stop and restart the application server for the changes to take effect.

---

**Deploying with Tomcat**

This section provides information used to deploy and configure MicroStrategy Web Services J2EE in a Tomcat-only environment. It assumes you have Tomcat deployed and operational. For information on how to configure Tomcat to work with IIS, refer to your Tomcat documentation.

For a list of application servers that are certified to work with MicroStrategy Web Services J2EE, see *MicroStrategy Web Services J2EE software requirements and recommendations, page 64.*

This section provides the following information:

- *Preconfiguration information, page 321:* configuration that must occur before you begin deploying MicroStrategy Web Universal
• *Deploying MicroStrategy Web Services J2EE, page 321*: instructions for deploying, including step-by-step procedures

**Preconfiguration information**

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Services J2EE on Tomcat.

**Locating the WAR file**

The MicroStrategy Web Services J2EE application is packaged within a single file, called a WAR (Web ARchive) file, following the J2EE specifications, which you must deploy to run the application in your application server environment. For more information on installation on UNIX or Linux, see *Installation procedures on UNIX and Linux, page 112*. For more information on installation on Windows, see *Installation procedure, page 92*.

To deploy the WAR file, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this section.

**Deploying MicroStrategy Web Services J2EE**

Assuming you have made all the necessary configurations described above, you can begin deploying MicroStrategy Web Services J2EE with Tomcat. This involves the following steps:

1. *Deploying using Tomcat as a stand-alone Web container, page 322*

2. *Viewing the MicroStrategy Web Services home page, page 323*
Deploying using Tomcat as a stand-alone Web container

The procedure below assumes you are deploying MicroStrategy Web Services J2EE with Tomcat for the first time. If you are redeploying Web Services, you must first delete the MicroStrategyWS directory within the webapps Tomcat directory.

To deploy MicroStrategy Web Services J2EE using Tomcat as a stand-alone Web container

1. Copy the MicroStrategyWS.war file to the TomcatDirectory\webapps directory. Instructions on locating the MicroStrategyWS.war file can be found in the section Locating the WAR file, page 321 above.

2. Start Tomcat from the system tray. Right-click on the Tomcat icon in the system tray and select Start service.

You can also start Tomcat from the command line. For instructions to start and stop Tomcat from the command line refer to the steps below.

Stop and start Tomcat from the command line

The steps to stop and start Tomcat may not work for your version and configuration of Tomcat. If the following steps do not work, refer to your vendor documentation on how to stop and start Tomcat.

3. From the Start menu, select Run. The Run dialog box opens.

4. Type cmd in the Open drop-down list and click OK. The command prompt opens.

5. Browse to the TomcatDirectory\bin directory, where TomcatDirectory is the folder in which you installed Tomcat. For example, in the command prompt, type:

   cd C:\Tomcat\bin

   where C:\Tomcat is the folder where you installed Tomcat. Press ENTER.

   If you installed Tomcat under the Program Files folder, type Progra~1 when you change folders in the command prompt, for
example, C:\Program Files\Tomcat\bin. Otherwise, the system does not recognize the folder.

6 Type **startup** to start Tomcat and **shutdown** to stop it.

MicroStrategy Web Services J2EE is deployed automatically. A **MicroStrategyWS** folder now exists under the **TomcatDirectory\webapps** directory.

**Viewing the MicroStrategy Web Services home page**

After configuring MicroStrategy Web Services with Tomcat you can access the MicroStrategy Web Services home page.

---

**To configure the MicroStrategy Web Services home page**

1 In a Web browser, access the MicroStrategy Web Services home page by specifying the following URL:

   http://localhost:port/MicroStrategyWS/

   The names are case-sensitive. If the application server is enabled with security, a dialog box related to the administrator authentication opens.

   If you are using Tomcat integrated with IIS, you do not need to specify the port number in the URL. However, when using Tomcat as a stand-alone Web container, you must specify the port number. The default port for Tomcat is 8080.

2 When prompted for a user name and password, use the same values you specified in the **tomcat-users.xml** file.

3 Once you log in, the MicroStrategy Web Services home page is displayed.

You have completed the steps required to deploy MicroStrategy Web Services J2EE. To validate access to the required runtime files, edit your project sources file, and test your MicroStrategy Web Services J2EE installation, refer to the section *Completing your MicroStrategy Web Services J2EE deployment, page 331*. 

© 2011 MicroStrategy, Inc.
Deploying with SAP

This section provides information used to deploy and configure MicroStrategy Web Services J2EE using the SAP NetWeaver Application Server 7.1.

This section includes the following information:

- **Preconfiguration information, page 324**: Configuration that must occur before you begin deploying MicroStrategy Web Services J2EE.
- **Deploying MicroStrategy Web Services J2EE, page 325**: Instructions for deploying the application.

Preconfiguration information

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Services J2EE on your machine. It provides setup information for the machine where the application server and Web server are already installed.

For information on the version numbers certified and supported by MicroStrategy, see the MicroStrategy readme file.

Locating the WAR file

The MicroStrategy Web Services J2EE application is packaged within a single file, called a WAR (Web ARchive) file, following the J2EE specifications. You must deploy this WAR file to run the application in your application server environment.

The MicroStrategy Web Services J2EE WAR file (MicroStrategyWS.war) is located in the MicroStrategy Web Services J2EE deployment directory you specified during installation. For more information on installation on UNIX or Linux, see Installation procedures on UNIX and Linux, page 112. For more information on installation on Windows, see Installation procedure, page 92.
To deploy the WAR file, you must follow a set of steps that are specific to the application server you are using. For more details, see the application server vendor documentation or follow the instructions within this guide. You should also follow any relevant preconfiguration instructions within this section.

After deploying the MicroStrategyWS.war file, you can view the directory contents including WEB-INF folder.

**Deploying MicroStrategy Web Services J2EE**

Once your machine has the necessary settings configured, you can deploy MicroStrategy Web Services J2EE. Deployment involves the following steps:

1. *Deploying MicroStrategy Web Services J2EE with the SAP NetWeaver Application Server, page 325*
2. *Viewing the MicroStrategy Web Services home page, page 327*

**Deploying MicroStrategy Web Services J2EE with the SAP NetWeaver Application Server**

Follow the steps provided in this section to deploy MicroStrategy Web Services J2EE as a WAR file.

---

**To deploy MicroStrategy Web Services J2EE as a WAR file**

1. Copy the MicroStrategyWS.war file to the Windows machine hosting your application server. The location in which you store the file is used later and referred to as path_to_war_file.

2. From the Windows Start menu, select Run. The Run dialog box opens.

3. In the Open drop-down list, type cmd, and click OK. A command prompt opens.

4. Using the command prompt, browse to the following directory within the SAP Application Server installation directory:

   /usr.sap/SID/Instance_Number/j2ee/deployment/scripts/
The SID and Instance_Number parameters are defined during installation and configuration of the SAP Application Server. The port number above refers to the P4 port number. The default port number is 50004.

5 Type the following command and press ENTER to deploy the MicroStrategyWS.war file:

```
Deploy.bat
user_name:password@localhost:port_number
path_to_war_file
```

The user name and password must have administrative access. The port number above refers to the P4 port number. The default port number is 50004.

6 Access the NetWeaver web admin console using the following URL:

```
http://localhost:port_number/nwa
```

The port number above refers to the J2EE engine port number. The default port number is 50000.

7 Login as an administrative user.

8 From the Operation Management tab, select the Systems tab, and then click Start & Stop.

9 Select Java EE Applications. A list of applications deployed on the application server are displayed.

10 Select the MicroStrategyWS application just deployed from the list.

11 From the Application Details section, select the Status tab, and then click Start.

12 Select On all instances and Set "Started" as Initial State.

When application is started, the status is displayed as Started.
Viewing the MicroStrategy Web Services home page

To view the MicroStrategy Web Services home page

1. Access the MicroStrategy Web Services home page by specifying the following URL in the Web browser:

   http://machinename:portnumber/MicroStrategyWS/

   The login dialog box opens.

2. Specify a user name and password.

Deploying with Oracle 10g

This section provides information used to deploy and configure MicroStrategy Web Services J2EE on the Windows 2003 SP1 operating system with Apache as the Web server and Oracle 10g (10.1.3) as the application server.

This section contains the following information:

- Preconfiguration information, page 327: Configuration that must occur before you begin deploying MicroStrategy Web Services J2EE.
- Deploying MicroStrategy Web Services J2EE, page 328: Instructions for deploying, including detailed steps.

Preconfiguration information

This section provides the preconfiguration information necessary to deploy MicroStrategy Web Services J2EE on your machine.

Locating the WAR file

The MicroStrategy Web Services J2EE application is packaged within a single file, called a WAR (Web ARchive) file, following the J2EE specifications. You must deploy this file to run the application in your application server environment.
The MicroStrategy Web Services J2EE WAR file (MicroStrategyWS.war) is located in the path you specified when installing MicroStrategy Web Services. For more information on installation on UNIX or Linux, see Installation procedures on UNIX and Linux, page 112. For more information on installation on Windows, see Installation procedure, page 92.

To deploy the MicroStrategyWS.war file, you must follow a set of steps that are specific to the application server you are using. For more details, see your application server vendor documentation or follow the instructions within this guide.

After deploying the MicroStrategyWS.war file, you can view the WEB-INF folder which contains subfolders named classes and lib. The log files are created and maintained in the application server root folder.

### Deploying MicroStrategy Web Services J2EE

After your machine is configured, you can start the deployment of MicroStrategy Web Services J2EE with Oracle 10g.

To deploy MicroStrategy Web Services J2EE, perform the following procedures:

1. [Deploying using the Oracle Enterprise Manager, page 328](#)

2. [Accessing the MicroStrategy Web Services home page, page 330](#)

### Deploying using the Oracle Enterprise Manager

You can access Oracle Enterprise Manager from the following URL:

http://machine_name:port_number/em

where `machine_name` is the machine name or IP address of the Oracle machine, and `port_number` is the port number of Oracle Enterprise Manager.
To deploy with the Oracle Enterprise Manager

1. Start the Apache Web Server. From the Start menu, point to OracleAS 10g - DEFAULT_HOME1, and then choose Start ApplicationServerName.MachineName.Domain.

2. To verify that the Apache Web Server has started, open Oracle Enterprise Manager, select HTTP Server, and then click Start.

3. Select the OC4J instance where you want to deploy MicroStrategy Web Services J2EE. This procedure assumes you are using the default instance name home. Click home. The OC4J: home page opens.

4. Select the Applications tab.

5. Click Deploy. The Deploy: Select Archive page opens.

6. In the Archive area, select Archive is present on local host.

7. Click Browse to navigate to and select the MicroStrategyWS.war file.
   For more information on locating the MicroStrategyWS.war file, see Locating the WAR file, page 327.

8. In the Deployment Plan area, select Automatically create a new deployment plan and click Next. The Deploy: Application Attributes page opens.

9. Enter the Application Name and Context Root. This procedure uses MicroStrategyWS as the Application Name and /MicroStrategyWS as the Context Root.

10. Click Next. The Deploy: Deployment Settings page opens.

   To configure class loading

11. In the Configure Class Loading task name, click the Go To Task (pencil) icon.

12. Enable Search Local Classes First for MicroStrategy Web Services J2EE.

13. Click OK. You are returned to the Deploy: Deployment Settings page.
To map a user to the admin security role

The MicroStrategy Web Services home page is accessible only to users with an admin security role. To control access to it, map the admin security role to only those users for whom you want to grant permission to work in the MicroStrategy Web Services home page. In Oracle 10g, the security users and groups are defined in the Oracle Enterprise Manager.

14 In the Map Security Roles task name, click the Go To Task (pencil) icon. The Deployment Settings: Map Security Roles page opens.

15 For the admin security role, select the Map Role (pencil) icon. The Deployment Settings: Map Security Role: admin page opens.

16 Select Map selected users and groups to this role.

17 In the Map Role to Users area, in the User text box, enter the user name to map to the admin security role. Click Add.

Repeat this step to add all users for whom you want to grant permission to work in the MicroStrategy Web Services home page.

18 Click Continue, and then click OK. You are returned to the Deploy: Deployment Settings page.

19 Click on Deploy to deploy the MicroStrategy Web Services application.

20 Stop and restart the Apache Web Server.

Any users mapped to the admin security role can now access MicroStrategy Web Services home page. For information on accessing this page, see Accessing the MicroStrategy Web Services home page below.

Accessing the MicroStrategy Web Services home page

To access the MicroStrategy Web Services home page

1 In a Web browser, access the MicroStrategy Web Services home page by specifying the following URL:

http://IP_address/MicroStrategyWS/

where IP_address is the IP address of the Oracle machine. Names are case-sensitive, so be sure to use the correct case when typing the URL.
When prompted for a user name and password, specify the values you used earlier when creating the user mapped to the security role. For more information, refer to the steps described in *Preparing for the application installation, page 305.*

After you log in, the MicroStrategy Web Services home page appears.

You have completed the steps required to deploy MicroStrategy Web Services J2EE. To validate access to the required runtime files, edit your project sources file and test your MicroStrategy Web Services J2EE installation, refer to the section *Completing your MicroStrategy Web Services J2EE deployment* below.

---

## Completing your MicroStrategy Web Services J2EE deployment

You must complete a few more steps before using your MicroStrategy Web Services J2EE deployment. The following steps are described at a high level in this section:

- *Validating your MicroStrategy Web Services J2EE deployment, page 331*
- *Editing the project sources file, page 332*
- *Verify Web Services connectivity with Intelligence Server, page 333*

This section assumes you have successfully deployed MicroStrategy Web Services J2EE and have been able to view the MicroStrategy Web Services home page.

For more details on the steps that are described in this section, or to troubleshoot problems that occur, refer to the *MicroStrategy Web Services Administration Guide.*

---

## Validating your MicroStrategy Web Services J2EE deployment

This step ensures that the application can load all of the JAR files required by MicroStrategy Web Services J2EE.
To validate your installation

1. On the home page, click Validate. The Axis Happiness Page is displayed.

2. The two sections at the top, Needed Components and Optional Components, report the status of individual JAR files.

If any Needed Components are missing or incompatible, you must correct the situation before proceeding. If you browse down the page, you can view more details about the environment.

Editing the project sources file

The project sources file provides connection information used by some applications, such as MicroStrategy Office. This section shows how to add a new project source to this file.

To edit the project sources file

1. Browse to the projectsources.xml file, which is located in the application root directory.

2. Edit projectsources.xml and define a new connection to your Intelligence Server machine. You must indicate a name for your project source, the machine name or IP address, the port number used by the Intelligence Server for listening, and the authentication mode. The following is an example of the syntax to define a connection to your Intelligence Server machine:

```xml
<ProjectSource>
    <ProjectSourceName>MicroStrategy Tutorial</ProjectSourceName>
    <ServerName>IntelligenceServerName</ServerName>
    <AuthMode>MWSStandard</AuthMode>
    <PortNumber>0</PortNumber>
    <MaxPooledConnections>100</MaxPooledConnections>
</ProjectSource>
```
Verify Web Services connectivity with Intelligence Server

To verify the Web Services configuration and connectivity to your Intelligence Server, the TestService operation can be run from your web browser. This operation uses Web Services operations to request a list of project sources using Web Services. It then uses the parameters that you supply to get a list of projects and connect to a project on your Intelligence Server.

To execute the TestService operation

1. Open the MicroStrategy Web Services home page.

   At the bottom of the page, a form for parameters used in the TestService operation is displayed. Above the form are instructions for using the form.

2. Enter the values for the parameters:

   • For the Port input parameter, if the default MicroStrategy Intelligence Server port is used then you can leave the field blank. Otherwise, enter the specific port number.

   • For the ProjectName input parameter, you can enter the name of your project. For example, MicroStrategy Tutorial is the name of the Tutorial project.

   • For the Login and the Password input parameter, you can enter the login name and password you use to log in to the project.

If the test is successful, you see a message that summarizes the results of GetProjectSources, GetProjects and Connect. If the test fails, you see an error message. A failed test must be corrected before you proceed.

Administering your MicroStrategy Web Services deployment

You can configure and manage MicroStrategy Web Services ASP.NET and J2EE using web pages and xml configuration pages on the server. For more information about administering MicroStrategy Web, see the MicroStrategy Web Services Administration Guide.
Configuring your MicroStrategy installation

After completing the steps to deploy MicroStrategy Web Services ASP.NET or J2EE, you can continue your setup and configuration. To help guide the rest of your installation and configuration steps, refer to the section *Installation and configuration checklists, page 87*, for installation and configuration checklists.
8

SETTING UP DOCUMENTS AND HTML DOCUMENTS

Introduction

This chapter explains the setup required for the Intelligence Server Universal to execute HTML documents and Report Services documents on UNIX and Linux platforms.

This chapter includes the following sections:

- Prerequisites, page 336
- Executing documents and HTML documents in UNIX/Linux, page 336
  - Setup for creating and executing HTML documents, page 337
  - Setup for executing existing HTML documents, page 341
  - Setup for executing Report Services documents, page 342
Prerequisites

This chapter assumes the following:

• You are familiar with UNIX or Linux.
• You are familiar with MicroStrategy Desktop and MicroStrategy
  Intelligence Server Universal.
• You are familiar with MicroStrategy HTML documents and Report
  Services documents.
• You have a Report Services product license if you are using Report
  Services documents. HTML documents do not require a Report Services
  product license.
• You have installed MicroStrategy Desktop on a Windows machine.
• You have installed MicroStrategy Intelligence Server Universal on a
  UNIX or Linux machine.

Note the following:

• Some of the steps described in this document may require root
  access permissions.
• You must perform extra configuration steps to allow Report
  Services documents to support non-Western European fonts on a
  UNIX system. For more information, see Graph and document
  support of non-Western European fonts, page 530 of Appendix B,
  Troubleshooting.

Executing documents and HTML documents in
UNIX/Linux

A MicroStrategy Report Services document contains objects representing
data coming from one or more reports, as well as positioning and formatting
information. Report Services documents help format data from multiple
reports in a single display and can be used for presentations. When you
create a document, you can specify the data that appears, control the layout,
formatting, grouping, and subtotaling of data and specify the position of page breaks. In addition, you can insert pictures and draw borders in the document.

In this chapter, the term document signifies a Report Services document. For additional information on Report Services documents, refer to the MicroStrategy Document Creation Guide.

An HTML document is a container for formatting, displaying, and distributing multiple reports on the same page, or at the same time within a project. You can create dashboards and scorecards to display a group of reports within the MicroStrategy platform.

HTML documents are created using MicroStrategy Desktop. Before creating or executing HTML documents, you must specify the HTML document directory using the Project Configuration dialog box in MicroStrategy Desktop. The HTML document directory stores HTML templates that are required by the MicroStrategy Intelligence Server Universal for executing HTML documents. You can store the HTML document directory on a UNIX or Linux platform, but you must share the directory with the Windows platform that includes MicroStrategy Desktop. For more information on setting up the HTML document directory on a UNIX or Linux platform, see Setup for creating and executing HTML documents, page 337.

For additional information on HTML documents, see the HTML Documents chapter in the MicroStrategy Advanced Reporting Guide.

Setup for creating and executing HTML documents

HTML documents can only be created with MicroStrategy Desktop on a Windows platform, but they can be stored and executed from a directory within a UNIX or Linux platform. The directory that stores the HTML documents must be accessible on the computer with Intelligence Server Universal and the Windows computer with Desktop.

Using the Project Configuration dialog box in MicroStrategy Desktop, you must specify the location of the HTML document directory as an absolute path. This document directory can be on a local machine or on a remote machine. Users require appropriate read and write permissions to access this directory. When MicroStrategy Intelligence Server Universal executes HTML documents, it requires read permission to the HTML document directory to access the HTML files.
For the procedure of setting up an HTML document directory between Windows and UNIX and Linux computers below, the following assumptions are made:

- You have installed MicroStrategy Desktop on a Windows computer and installed MicroStrategy Intelligence Server Universal on a UNIX or Linux computer.
  
  Desktop can only be installed on a Windows computer.

- MicroStrategy Desktop users have at least read permissions to the HTML document directory for executing existing HTML documents. Write permissions to the directory are required for MicroStrategy Desktop users to create new HTML documents.

- For the file paths described in the procedure below, `machine-name` is used to represent the name of the machine you store the HTML document directory on. For example, if you store the directory on a machine named UNIX1, `machine-name` should be replaced with UNIX1. This machine must have Samba installed to provide access to the folder on a Windows computer.

- You must have root permissions on any UNIX or Linux computer used to set up the HTML document directory. This includes the computer that stores the HTML document directory as well as any computer that must be setup to access the directory.

---

**To set up the HTML document directory**

1. Create a directory to hold the HTML document directory on the desired UNIX or Linux computer. This procedure assumes that the path of the HTML document directory is `machine-name:/share/htmldocuments`. This is the machine that is referenced as `machine-name` in the steps below. To create this directory, enter the commands below:

   ```bash
   # cd /
   # mkdir share
   # cd share
   # mkdir htmldocuments
   ```

2. Install Samba software on the UNIX or Linux computer that you created the HTML documents directory in the step above. With this software, the HTML documents directory is accessible to the Windows computer with

Notice that Samba uses a .org extension and not the more common .com extension. Using a .com extension takes you to an incorrect website.

3 Share the directory `machine-name:/share` across the network through NFS. For example, you must share `UNIX1:/share`. Make sure read and write permissions are set for the share. This step allows other UNIX/Linux computers to access the directory.

4 Create a Samba share, named “share”, with read and write permissions that points to the directory `machine-name:/share`. For example, you must share `UNIX1:/share`. This step allows Windows computers to access the directory.

The Samba share is created in the Samba `smb.conf` file. For specific instructions on how to setup a Samba share, refer to the Samba website at http://www.samba.org.

5 Restart Samba.

6 Mount the HTML document directory on the computer that has the Intelligence Server installed on it. Root privileges are required for this.

On the computer with Intelligence Server, type the command `su` and the root password at the command prompt to log in as a superuser, or log in as `root`. The command prompt changes to the pound sign (#). Perform the commands below:

In the commands below, `machine-name` refers to the machine name of the computer where you stored the HTML documents directory and created an NFS and Samba share. This may be a different name than the computer that you are mounting the directory on.

The final mount command contains a space between `/htmldocuments` and `/machine-name`.

```
cd /
mkdir machine-name
```
mkdir htmldocuments

cd /

mount machine-name:/share/htmldocuments
/machine-name/share/htmldocuments

7 You can cache the connection to the UNIX/Linux HTML documents directory from the Windows computer so that you are not prompted for authentication each time the directory is accessed:

a From the Windows computer that has MicroStrategy Desktop installed, click Start, and select Run. The Run dialog box opens.

b Type `\machine-name\share\htmldocuments`, and click OK to open the top-level shared HTML documents directory. For example, type `\UNIX1\share\htmldocuments`.

This must be performed every time you restart the computer.

8 Using the Project Configuration dialog box in MicroStrategy Desktop, set the HTML document directory as an absolute path by following the steps below:

a In Desktop, right-click the project associated with the HTML documents and select Project Configuration. The Project Configuration dialog box opens.

b Expand Project definition and click Advanced. The Project Configuration - Advanced options are displayed.

c In the HTML document directory box, type the absolute path `\machine-name\share\htmldocuments`. For example, type `\UNIX1\share\htmldocuments`.

d Click OK to accept the changes.

9 Create a directory named xsls under the HTML document directory and copy the XSL files you require for creating HTML documents to the xsls directory, `/machine-name/share/htmldocuments/xsls`. If you stored XSL files in a different directory or did not copy them from their original default directory, you must copy them into the new xsls directory. For example, the default HTML document directory for the Tutorial project is `Program Files\MicroStrategy\Tutorial Reporting`.

10 If you want to insert images into the HTML document, create a directory named images under the HTML document directory, and copy the
images to the directory
/machine-name/share/htmldocuments/images.

You are now ready to create and execute your HTML documents. Remember to create your HTML documents in the HTML document directory, otherwise, Intelligence Server cannot execute the HTML documents correctly.

Setup for executing existing HTML documents

If you have created HTML documents prior to establishing a connection between the HTML document directory on the UNIX or Linux machine with MicroStrategy Intelligence Server and the Windows machine with MicroStrategy Desktop, you must make sure that all the files used for an HTML document are copied to the shared HTML document directory. After the connection is established, you should always create the HTML documents in the shared HTML document directory. Once the existing files are copied, you can execute the HTML documents using Intelligence Server.

In the procedure of setting up existing HTML documents, the following assumptions are made:

- You have completed all the steps listed in the section Setup for creating and executing HTML documents, page 337.
- The location of the HTML document directory is /machine-name/share/htmldocuments. For the file paths described in the procedure below, machine-name is used to represent the name of the machine on which you store the HTML document directory. For example, if you store the directory on a machine named UNIX1, machine-name should be replaced with UNIX1.

To set up existing HTML documents for execution

1. Copy the HTML file for any existing HTML document to /machine-name/share/htmldocuments.

2. View the source code of each HTML file and copy the XSL file used by each HTML document in an appropriate directory under /machine-name/share/htmldocuments.

   For example, if the location of the XSL file in the source code is xsl="\xsls\myxsl.xsl, then copy myxsl.xsl to
/machine-name/share/htmldocuments/xsls. If the location of the XSL file in the source code is xsl="\myxsl.xsl, then copy myxsl.xsl to /machine-name/share/htmldocuments.

3 View the source code for the images used by each HTML document in an appropriate directory under
/machine-name/share/htmldocuments.

For example, if the location of the image file in the source code is \images\myimage.gif, then copy myimage.gif to
/machine-name/share/htmldocuments/images. If the location of the XSL file in the source code is \myimage.gif, then copy myimage.gif to /machine-name/share/htmldocuments.

You are now ready to execute your HTML documents.

**Setup for executing Report Services documents**

A MicroStrategy Report Services document is used to format data from multiple reports. These documents can be exported to PDF format. To execute documents and export them to PDF format using MicroStrategy Intelligence Server Universal in a UNIX or Linux environment, you must perform some additional setup tasks.

When Intelligence Server Universal is running on a UNIX or Linux platform, all fonts are converted to the Courier New font for:

- Reports exported to PDF format
- Report Services documents
- Graphs contained in HTML documents
- Graphs displayed in MicroStrategy Web

This occurs because the fonts required by the PDF component are missing from UNIX and Linux machines running Intelligence Server Universal. The missing fonts may include Microsoft True Type fonts.

MicroStrategy does not distribute or license Microsoft fonts, and therefore cannot package Microsoft fonts with Intelligence Server Universal.

To resolve this issue, you must install the font files in the PDFGeneratorFiles folder within the MicroStrategy installation path on the UNIX or Linux machine, as described below.
To copy fonts to your UNIX or Linux machine

1 Log in to your UNIX or Linux machine that hosts Intelligence Server Universal.

2 Install the Microsoft True Type fonts. Refer to the following resources for information on licensing requirements for and installing Microsoft True Type fonts:
   • http://www.microsoft.com/typography/RedistributionFAQ.mspx
   • http://www.ascendercorp.com/msfonts/msfonts_main.html
   • http://corefonts.sourceforge.net/

3 Copy the font files into the INSTALL_PATH/PDFGeneratorFiles directory, where INSTALL_PATH is the directory you specified as the MicroStrategy install directory during installation.

4 To update the list of fonts available, you must restart the Intelligence Server.

Configuring your MicroStrategy installation

After completing the steps to set up documents and HTML documents for UNIX and Linux, you can continue your setup and configuration. To help guide the rest of your installation and configuration steps, refer to the section Installation and configuration checklists, page 87 in Chapter 1, Planning Your Installation, for installation and configuration checklists.
Introduction

This chapter explains the various possibilities for performing fully automated and unattended installations within the MicroStrategy platform. This includes customizations to the installation routines available with the product. It explains the different resources you can use to deploy MicroStrategy products through various scenarios including:

• Deploying the MicroStrategy platform across the network through the Microsoft System Management Server (SMS) or its equivalent (for example, IBM Tivoli)

• Embedding the MicroStrategy platform within third party custom applications and other installation routines

• Customizing the MicroStrategy installation to meet the various environment-specific requirements for a given site

This chapter provides the following information:

• The different types of installations that can be performed with MicroStrategy products.
• How to perform a fully automated MicroStrategy installation by modifying various installation parameters in Windows ini-like response files.

• How to customize certain aspects of the MicroStrategy installation to meet various site-specific requirements for multi-user environments, such as strict standards for software deployment to user communities and so on.

Automated and silent installations require advanced techniques such as creating and running response.ini files. Therefore, automated and silent installations should be handled by system administrators with full knowledge of the environment and the desired MicroStrategy installation.

Before installing MicroStrategy products, you should refer to Chapter 1, Planning Your Installation for important pre-installation information.

Installation log file

Before you begin to learn about automated installation options, it is important to know about the installation log file. The setup program generates a log file in text format. This log file contains records of all actions performed by the setup program and by other executable files related to installation. The installation log file can be particularly helpful if you encounter errors during the installation process. For example, the log can tell you if a registry key or path was not added or if a critical file was not registered successfully. The setup.exe file writes to this log file. The log file data includes:

• Update dates
• Machine specifications
• User selections
• List of files to be registered
• List of files that do not require registration
• List of registry entries
• Identification of files that fail during registration
• Installation activity such as performance counter loading and DSN creation
Reboot time file registration results

The default location for the install.log file is:

- **32-bit Windows environments**: Program Files\Common Files\MicroStrategy
- **64-bit Windows environments**: Program Files (x86)\Common Files\MicroStrategy

Both the location and the name can be changed. You can specify the log file name and location in the following places:

- Command line, reading the parameter **LogFile**. For example:
  ```
  setup.exe --LogFile="C:\install.log"
  ```
- Response file in [LogFile]. See *Configuring a response.ini file to install MicroStrategy, page 348* for more information.

## Methods of installation

The installation methods discussed in this chapter are:

- *Installing and configuring with a response.ini file, page 347*
- *Silent installation, page 398*

## Installing and configuring with a response.ini file

The response.ini file can facilitate the installation and setup of MicroStrategy products by allowing you to progress through the installation and project creation processes with a single keystroke. A response.ini file is an initialization file that is used to send parameters or selections to the MicroStrategy Installation Wizard. This allows you to run it silently as all the options are pre-selected in that file. This section describes how to create and use response.ini for the following tasks:

- *Configuring a response.ini file to install MicroStrategy, page 348*
- *Configuring your installation with a response.ini file, page 396*
Configuring a response.ini file to install MicroStrategy

The response.ini file for installation allows you to automate certain aspects of the installation by modifying a Windows ini-like response file. This option is typically implemented by:

- OEM applications that embed MicroStrategy installations within other products
- IT departments who want to have more control over desktop installations

The response.ini file specifies all the selections you want to make during the installation in the MicroStrategy Installation Wizard. You can either run it with all the MicroStrategy Installation Wizard options that are pre-selected, or run it without having to use the wizard at all.

- The response.ini file should not be confused with the setup.iss file, which is used by the MicroStrategy Installation Wizard to perform silent installation. When both response.ini and setup.iss are included in the setup, response.ini overrides setup.iss. For details on the setup.iss file, see Silent installation, page 398.

Component dependencies

When you use a response.ini file to install MicroStrategy products, there are some key dependencies among separate components you should be aware of. The products listed below require either pre-installed software or certain MicroStrategy components to be selected to successfully install the products with a response.ini file:

- MicroStrategy Narrowcast Server Administrator requires SequeLink ODBC Socket Server to be included in the installation
- MicroStrategy Narrowcast Server Subscription Portal requires SequeLink ODBC Socket Server to be included in the installation
- MicroStrategy Function Plug-in Wizard requires Microsoft Visual C++ to be installed before running your response.ini file
- MicroStrategy Analytic Modules requires MicroStrategy Desktop Designer or a combination of MicroStrategy Desktop Analyst and MicroStrategy Architect to be included in the installation
Creating a response.ini file

You can create a response.ini file in any text editor and save the file as response.ini in the desired folder.

⚠️ You must save the file as ANSI encoding.

The following tables describe the parameters and options for the all the sections, such as Installer, Paths, and so on in the response.ini file. It is followed by sample response.ini files for your reference.

⚠️ The options are case sensitive, therefore they must be entered as indicated in the tables below.

### Installer

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Installer]</td>
<td>Section that begins the installation.</td>
</tr>
<tr>
<td>HideAllDialogs =</td>
<td>TRUE or FALSE. Indicates whether the installation uses all default values. FALSE displays all the dialog boxes and you must browse using the Next buttons. The default is FALSE.</td>
</tr>
</tbody>
</table>
| LogFile =                | Location where the install.log file is generated. If left empty, it takes the default location and file name of:  
  • 32-bit Windows environments: C:\Program Files\Common Files\MicroStrategy\ install.log.  
  • 64-bit Windows environments: C:\Program Files (x86)\Common Files\MicroStrategy\install.log. |
| CreateShortcuts =        | TRUE or FALSE. TRUE creates a shortcut called MicroStrategy in the Windows Start menu. The default is TRUE. |
| PreventReboot =          | TRUE or FALSE. TRUE prevents the machine from rebooting after installation is done. Note the following conditions:  
  • If both ForceReboot = TRUE and PreventReboot = TRUE, then PreventReboot applies first.  
  • If both ForceReboot and PreventReboot are FALSE and HideDialog for [Finish] is set to TRUE, then the machine reboots only if it is required.  
  The default is FALSE. |
<p>| ForceReboot =            | TRUE or FALSE. TRUE reboots the machine after the installation is done. The default is FALSE. |
| EnterpriseManagerOverwrite = | TRUE or FALSE. If you select FALSE, the warehouse and metadata files are not updated but the rest of the files for Enterprise Manager are updated. This prompt only impacts the files in the Microsoft Access database. The default is FALSE. |</p>
<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ShowConfigWizard =</td>
<td>TRUE or FALSE. When using silent install, set to FALSE to prevent the Configuration Wizard from coming up after reboot. The default is TRUE.</td>
</tr>
<tr>
<td>RunConfigWizard =</td>
<td>TRUE or FALSE. Set this option to specify whether to run the Configuration Wizard. The default is FALSE.</td>
</tr>
<tr>
<td>ConfigWizardResponseFile =</td>
<td>Specify the name of the response file for the Configuration Wizard; otherwise, it takes the default name of response.ini. For more details on configuring the response.ini file for the Configuration Wizard, see Configuring your installation with a response.ini file, page 396.</td>
</tr>
<tr>
<td>BackupFiles =</td>
<td>TRUE or FALSE. If you set the value to TRUE, it creates a backup of the following files: * .pds, * .xsl, * .asp, * .css, * .js, * .sql. The default is FALSE.</td>
</tr>
<tr>
<td>StopAllServices =</td>
<td>TRUE or FALSE. Set this option to stop all MicroStrategy services that are running on the machine during installation. The default is FALSE.</td>
</tr>
<tr>
<td>StopIIS =</td>
<td>TRUE or FALSE. Set this option to stop Internet Information Services (IIS) during installation. The default is FALSE.</td>
</tr>
<tr>
<td>AnalyticsOverwrite =</td>
<td>TRUE or FALSE. This option overwrites the Analytics Modules from a previous install. The default is FALSE.</td>
</tr>
<tr>
<td>TutReportingOverwrite =</td>
<td>TRUE or FALSE. Set this option to overwrite the Reporting Tutorial from a previous installation. The default is FALSE.</td>
</tr>
<tr>
<td>WebMMTForWebOverwrite =</td>
<td>TRUE or FALSE. Set this option to overwrite the MicroStrategy Web MMT database from a previous installation. The default is FALSE.</td>
</tr>
<tr>
<td>TutDeliveryOverwrite =</td>
<td>TRUE or FALSE. Set this option to overwrite the Delivery Tutorial from a previous installation. The default is FALSE.</td>
</tr>
<tr>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| CheckRenameOperations =       | **TRUE** or **FALSE**. In some instances, as a result of a previous installation or an uninstall, certain files may be missing or irreplaceable during installation. Therefore, you are prompted to do one of the following:  
  • Reboot at the beginning of installation to replace this file. It is recommended that you select this option.  
  • Continue with the installation at the risk of the software not functioning properly.  
  If you enter **FALSE**, the prompt does not display. The default is **TRUE**. |
| EnableTracing =               | **TRUE** or **FALSE**. Set this option to trace the setup process in a log file that is saved in the Temp folder. The log file records errors that are encountered during the installation. The default is **FALSE**. |
| EnableASPNETServices =         | **TRUE** or **FALSE**. Set this option to enable the ASP.NET MicroStrategy Web Services extensions that IIS Admin requires. The default is **FALSE**. |
| EnableASPServices =            | **TRUE** or **FALSE**. Set this option to enable the ASP MicroStrategy Web Services extensions that IIS Admin requires. The default is **FALSE**. |
| ShowWelcomeScreen =           | **TRUE** or **FALSE**. Set to **TRUE** to display the Welcome screen after reboot. The Welcome screen is displayed only once after reboot. The default is **TRUE**. |
| PropertiesFilesOverwrite=     | **TRUE** or **FALSE**. Set to **TRUE** to create new properties files. These files are related to the Tutorial Delivery component. The default is **FALSE**, which uses the current version of the properties files. The default is **FALSE**. |
| CheckTCPIP=                   | **TRUE** or **FALSE**. Set to **TRUE** to check that the TCP/IP network protocol is active. If set to **FALSE**, the setup doesn’t check for it. The default is **TRUE**. |
| CheckIIS=                     | **TRUE** or **FALSE**. Set to **TRUE** to check for Internet Information Services. The default is **TRUE**. |
| StopAllServices=              | **TRUE** or **FALSE**. Set to **TRUE** to stop all services required to be stopped to complete a MicroStrategy installation. If set to **FALSE**, the user is prompted if services need to be stopped. The default is **TRUE**. |
### Paths

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[InitialPaths]</td>
<td>Section for specifying the path for the products that you select to install.</td>
</tr>
<tr>
<td>COMMONFILES</td>
<td>Location where the common files like <code>response.ini</code>, <code>install.log</code>, and so on will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>- 32-bit Windows environments: <code>C:\Program Files\Common Files\MicroStrategy</code></td>
</tr>
<tr>
<td></td>
<td>- 64-bit Windows environments: <code>C:\Program Files (x86)\Common Files\MicroStrategy</code></td>
</tr>
<tr>
<td>Desktop</td>
<td>Location where Desktop will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>- 32-bit Windows environments: <code>C:\Program Files\MicroStrategy\Desktop</code></td>
</tr>
<tr>
<td></td>
<td>- 64-bit Windows environments: <code>C:\Program Files (x86)\MicroStrategy\Desktop</code></td>
</tr>
<tr>
<td>ObjectManager</td>
<td>Location where Object Manager will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>- 32-bit Windows environments: <code>C:\Program Files\MicroStrategy\Object Manager</code></td>
</tr>
<tr>
<td></td>
<td>- 64-bit Windows environments: <code>C:\Program Files (x86)\MicroStrategy\Object Manager</code></td>
</tr>
<tr>
<td>CommandManager</td>
<td>Location where Command Manager will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>- 32-bit Windows environments: <code>C:\Program Files\MicroStrategy\Command Manager</code></td>
</tr>
<tr>
<td></td>
<td>- 64-bit Windows environments: <code>C:\Program Files (x86)\MicroStrategy\Command Manager</code></td>
</tr>
<tr>
<td>EnterpriseManager</td>
<td>Location where Enterprise Manager will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>- 32-bit Windows environments: <code>C:\Program Files\MicroStrategy\Enterprise Manager</code></td>
</tr>
<tr>
<td></td>
<td>- 64-bit Windows environments: <code>C:\Program Files (x86)\MicroStrategy\Enterprise Manager</code></td>
</tr>
<tr>
<td>Server</td>
<td>Location where the MicroStrategy Intelligence Server will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>- 32-bit Windows environments: <code>C:\Program Files\MicroStrategy\Intelligence Server</code></td>
</tr>
<tr>
<td></td>
<td>- 64-bit Windows environments: <code>C:\Program Files (x86)\MicroStrategy\Intelligence Server</code></td>
</tr>
<tr>
<td><strong>Options</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>-------------</td>
<td>----------------</td>
</tr>
</tbody>
</table>
| **Web =**   | Location where MicroStrategy Web will be installed. If left empty, it takes the default location of:  
  • 32-bit Windows environments: `C:\Program Files\MicroStrategy\Web ASPx`  
  • 64-bit Windows environments: `C:\Program Files (x86)\MicroStrategy\Web ASPx` |
| **WebUniversal =** | Location where MicroStrategy Web Universal will be installed. If left empty, it takes the default location of:  
  • 32-bit Windows environments: `C:\Program Files\MicroStrategy\Web JSP`  
  • 64-bit Windows environments: `C:\Program Files (x86)\MicroStrategy\Web JSP` |
| **WebServices =** | Location where MicroStrategy Web Services will be installed. If left empty, it takes the default location of:  
  • 32-bit Windows environments: `C:\Program Files\MicroStrategy\Web Services`  
  • 64-bit Windows environments: `C:\Program Files (x86)\MicroStrategy\Web Services` |
| **WebServicesUniversal =** | Location where MicroStrategy Web Services Universal will be installed. If left empty, it takes the default location of:  
  • 32-bit Windows environments: `C:\Program Files\MicroStrategy\Web Services JSP`  
  • 64-bit Windows environments: `C:\Program Files (x86)\MicroStrategy\Web Services JSP` |
| **SDK =** | Location where the SDK will be installed. If left empty, it takes the default location of:  
  • 32-bit Windows environments: `C:\Program Files\MicroStrategy\SDK`  
  • 64-bit Windows environments: `C:\Program Files (x86)\MicroStrategy\SDK` |
| **Office =** | Location where MicroStrategy Office will be installed. If left empty, it takes the default location of:  
  • 32-bit Windows environments: `C:\Program Files\MicroStrategy\Office`  
  • 64-bit Windows environments: `C:\Program Files (x86)\MicroStrategy\Office` |
| **TutorialReporting =** | Location where MicroStrategy Tutorial - Reporting will be installed. If left empty, it takes the default location of:  
  • 32-bit Windows environments: `C:\Program Files\MicroStrategy\Tutorial Reporting`  
  • 64-bit Windows environments: `C:\Program Files (x86)\MicroStrategy\Tutorial Reporting` |
<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WebMMT =</td>
<td>Location where MicroStrategy Web MMT will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>• 32-bit Windows environments: C:\Program Files \MicroStrategy\WebMMT</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environments: C:\Program Files (x86) \MicroStrategy\WebMMT</td>
</tr>
<tr>
<td>AnalyticsModules =</td>
<td>Location where the Analytics Modules will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>• 32-bit Windows environments: C:\Program Files \MicroStrategy\Analytics Modules</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environments: C:\Program Files (x86) \MicroStrategy\Analytics Modules</td>
</tr>
<tr>
<td>NCSAdminDeliveryEngine =</td>
<td>Location where the Narrowcast Server Delivery Engine will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>• 32-bit Windows environments: C:\Program Files \MicroStrategy\Narrowcast Server\Delivery Engine</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environments: C:\Program Files (x86) \MicroStrategy\Narrowcast Server\Delivery Engine</td>
</tr>
<tr>
<td>SubscriptionPortal =</td>
<td>Location where the Subscription Portal will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>• 32-bit Windows environments: C:\Program Files \MicroStrategy\Narrowcast Server\Subscription Portal</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environments: C:\Program Files (x86) \MicroStrategy\Narrowcast Server\Subscription Portal</td>
</tr>
<tr>
<td>TutorialDelivery =</td>
<td>Location where MicroStrategy Tutorial - Delivery will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>• 32-bit Windows environments: C:\Program Files \MicroStrategy\Narrowcast Server\Tutorial</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environments: C:\Program Files (x86) \MicroStrategy\Narrowcast Server\Tutorial</td>
</tr>
<tr>
<td>IntegrityManager =</td>
<td>Location where MicroStrategy Integrity Manager will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>• 32-bit Windows environments: C:\Program Files \MicroStrategy\Integrity Manager</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environments: C:\Program Files (x86) \MicroStrategy\Integrity Manager</td>
</tr>
<tr>
<td>Mobile =</td>
<td>Location where MicroStrategy Mobile will be installed. If left empty, it takes the default location of:</td>
</tr>
<tr>
<td></td>
<td>• 32-bit Windows environments: C:\Program Files \MicroStrategy\Mobile Clients</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environments: C:\Program Files (x86) \MicroStrategy\Mobile Clients</td>
</tr>
</tbody>
</table>
### Options

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
</table>
| MobileASPPath =  | Location where MicroStrategy Mobile Server ASP.NET will be installed. If left empty, it takes the default location of:  
• 32-bit Windows environments: C:\Program Files\MicroStrategy\Mobile Server ASPx  
• 64-bit Windows environments: C:\Program Files (x86)\MicroStrategy\Mobile Server ASPx |
| MobileJSPPath =  | Location where MicroStrategy Mobile Server JSP will be installed. If left empty, it takes the default location of:  
• 32-bit Windows environments: C:\Program Files\MicroStrategy\Mobile Server JSP  
• 64-bit Windows environments: C:\Program Files (x86)\MicroStrategy\Mobile Server JSP |
| Portlets =       | Location where MicroStrategy Portlets will be installed. If left empty, it takes the default location of:  
• 32-bit Windows environments: C:\Program Files\MicroStrategy\Portlets  
• 64-bit Windows environments: C:\Program Files (x86)\MicroStrategy\Portlets |
| TM1Connector =   | Location where the TM1 Connector for MicroStrategy will be installed. If left empty, it takes the default location of:  
• 32-bit Windows environments: C:\Program Files\MicroStrategy\TM1 Connector  
• 64-bit Windows environments: C:\Program Files (x86)\MicroStrategy\TM1 Connector |
| GISConnectors =  | Location where the MicroStrategy GIS Connectors will be installed. If left empty, it takes the default location of:  
• 32-bit Windows environments: C:\Program Files\MicroStrategy\GISConnectors  
• 64-bit Windows environments: C:\Program Files (x86)\MicroStrategy\GISConnectors |

### Welcome dialog box

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Welcome]</td>
<td>Section for configuring the Welcome dialog box.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
</tbody>
</table>
| RemoveAll =     | TRUE or FALSE. This option is for the uninstall process only. Setting it to TRUE removes all MicroStrategy products during the uninstall process. The default is FALSE.  
For an example of a response file used to uninstall all MicroStrategy products, see Uninstalling with a response.ini file, page 396. |
Customer Information dialog box

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[UserRegistration]</td>
<td>Section for specifying the customer information.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>UserName =</td>
<td>Indicates the user name of the currently logged user or a user who is already registered. If no information is provided, you cannot proceed to the next page.</td>
</tr>
<tr>
<td>CompanyName =</td>
<td>The name of the company for which the software is registered. The default is the company name in the registry.</td>
</tr>
<tr>
<td>LicenseKey =</td>
<td>Specify the license key for the software. If you do not specify the license key, the MicroStrategy Installation Wizard will ask for it when it reaches that step. By default it is blank for a fresh install or displays the license key from a previous install.</td>
</tr>
</tbody>
</table>

Setup Type dialog box

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[SetupType]</td>
<td>Section for setting the appropriate dialog boxes to display for Typical or Advanced installation. This dialog box offers the option to select the installation type.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>Type =</td>
<td>This determines whether the installation is Typical or Advanced. One of the following two parameters can be entered:</td>
</tr>
<tr>
<td></td>
<td>• typical</td>
</tr>
<tr>
<td></td>
<td>• advanced</td>
</tr>
<tr>
<td></td>
<td>Typical or Advanced. If set to typical, the installation paths specified in the [InstallPaths] section are used. If set to Advanced, you are prompted to enter the path for each product. For more information, refer to Graphical user interface, page 84 in Chapter 1, Planning Your Installation. The default is Advanced.</td>
</tr>
</tbody>
</table>

Choose Destination Location dialog box

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[SuiteTarget]</td>
<td>Section specifying the name of the target directory from where you can run the MicroStrategy products.</td>
</tr>
</tbody>
</table>
Select Components dialog box

In the MicroStrategy Installation Wizard, the Select Components dialog box contains check boxes to select or clear for products to be installed. The [ComponentSelection] options specify whether you want the following products to be visible to the user. In addition, you can set the default selection for each product.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>TargetDirectory =</td>
<td>Location of the root directory for the Program Files. The default is set to:</td>
</tr>
<tr>
<td></td>
<td>• 32-bit Windows environment: C:\Program Files\MicroStrategy</td>
</tr>
<tr>
<td></td>
<td>• 64-bit Windows environment: C:\Program Files (x86)\MicroStrategy</td>
</tr>
<tr>
<td>[ComponentSelection]</td>
<td>Equivalent to the Select Components dialog box that you see during installation. For the Visible option, you can either enter TRUE to show a product or FALSE to hide it. If you do not specify a TRUE or FALSE value for each product, TRUE is used for all products. For the Select option, you can enter TRUE to select the check box next to a product. If you enter FALSE, the check box next to the product is not selected.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>AnalyticsModulesVisible =</td>
<td>MicroStrategy Analytics Modules</td>
</tr>
<tr>
<td>ArchitectVisible =</td>
<td>MicroStrategy Architect</td>
</tr>
<tr>
<td>CommandManagerVisible =</td>
<td>MicroStrategy Command Manager</td>
</tr>
<tr>
<td>DeliveryEngineVisible =</td>
<td>MicroStrategy Delivery Engine</td>
</tr>
<tr>
<td>DesktopDesignerVisible =</td>
<td>MicroStrategy Desktop Designer</td>
</tr>
<tr>
<td>DesktopAnalystVisible =</td>
<td>MicroStrategy Desktop Analyst</td>
</tr>
<tr>
<td>EnterpriseManagerVisible =</td>
<td>MicroStrategy Enterprise Manager</td>
</tr>
<tr>
<td>FunctionPluginVisible =</td>
<td>MicroStrategy Function Plug-In Wizard</td>
</tr>
<tr>
<td>GISConnectorsVisible =</td>
<td>MicroStrategy GIS Connectors</td>
</tr>
<tr>
<td>IntegrityManagerVisible =</td>
<td>MicroStrategy Integrity Manager</td>
</tr>
<tr>
<td>IServerDistributionServicesVisible =</td>
<td>MicroStrategy Distribution Services</td>
</tr>
<tr>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>IServerOLAPServicesVisible =</td>
<td>MicroStrategy OLAP Services</td>
</tr>
<tr>
<td>IServerReportServicesVisible =</td>
<td>MicroStrategy Report Services</td>
</tr>
<tr>
<td>IServerTransactionServicesVisible =</td>
<td>MicroStrategy Transaction Services</td>
</tr>
<tr>
<td>IServerVisible =</td>
<td>MicroStrategy Intelligence Server</td>
</tr>
<tr>
<td>MobileClientVisible =</td>
<td>MicroStrategy Mobile for BlackBerry client</td>
</tr>
<tr>
<td>MobileServerASPVisible =</td>
<td>MicroStrategy Mobile Server (ASP.NET)</td>
</tr>
<tr>
<td>MobileServerJSPVisible =</td>
<td>MicroStrategy Mobile Server (JSP)</td>
</tr>
<tr>
<td>MobileVisible =</td>
<td>MicroStrategy Mobile</td>
</tr>
<tr>
<td>NCSAdminVisible =</td>
<td>MicroStrategy Narrowcast Administrator</td>
</tr>
<tr>
<td>ObjectManagerVisible =</td>
<td>MicroStrategy Object Manager</td>
</tr>
<tr>
<td>OfficeVisible =</td>
<td>MicroStrategy Office</td>
</tr>
<tr>
<td>PortletsVisible =</td>
<td>MicroStrategy Portlets</td>
</tr>
<tr>
<td>ProjectBuilderVisible =</td>
<td>MicroStrategy Project Builder</td>
</tr>
<tr>
<td>SDKVisible =</td>
<td>MicroStrategy SDK</td>
</tr>
<tr>
<td>SequeLinkVisible =</td>
<td>SequeLink ODBC Socket Server</td>
</tr>
<tr>
<td>ServerAdminVisible =</td>
<td>MicroStrategy Server Administrator</td>
</tr>
<tr>
<td>SubscriptionPortalVisible =</td>
<td>MicroStrategy Subscription Portal</td>
</tr>
<tr>
<td>TM1ConnectorVisible =</td>
<td>TM1 Connector for MicroStrategy</td>
</tr>
<tr>
<td>TutorialDeliveryConfigureVisible =</td>
<td>MicroStrategy Tutorial - Delivery Configuration</td>
</tr>
<tr>
<td>TutorialDeliveryInstallVisible =</td>
<td>MicroStrategy Tutorial - Delivery Installation</td>
</tr>
<tr>
<td>TutorialReportingVisible =</td>
<td>MicroStrategy Tutorial - Reporting</td>
</tr>
<tr>
<td>WebAnalystVisible =</td>
<td>MicroStrategy Web Analyst</td>
</tr>
<tr>
<td>WebMMTVisible =</td>
<td>MicroStrategy Web MMT</td>
</tr>
<tr>
<td>WebProfessionalVisible =</td>
<td>MicroStrategy Web Professional</td>
</tr>
<tr>
<td>WebReporterVisible =</td>
<td>MicroStrategy Web Reporter</td>
</tr>
<tr>
<td>WebServerASPNETVisible =</td>
<td>MicroStrategy Web Server (ASP.NET)</td>
</tr>
<tr>
<td>WebServerJSPVisible =</td>
<td>MicroStrategy Web Server (JSP)</td>
</tr>
<tr>
<td>WebServicesASPNETVisible =</td>
<td>MicroStrategy Web Services (ASP.NET)</td>
</tr>
<tr>
<td>WebServicesJSPVisible =</td>
<td>MicroStrategy Web Services Universal (JSP)</td>
</tr>
</tbody>
</table>
During the installation process in the MicroStrategy Installation Wizard, the Select Components dialog box contains check boxes to select or clear for products to be installed. You can either specify `TRUE` to install a product or `FALSE` to uninstall it. If you do not specify a `TRUE` or `FALSE` value for each product, the installation always uses the most recent selection from a previous install.

This means that if you have a product installed and you do not specify a `TRUE` or `FALSE` value, the product is upgraded.

If you specify `TRUE`, the product check box is selected. The `[ComponentSelection]` options specify whether the check box for each product will be selected or cleared.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AnalyticsModulesSelect =</td>
<td>MicroStrategy Analytics Modules</td>
</tr>
<tr>
<td>ArchitectSelect =</td>
<td>MicroStrategy Architect</td>
</tr>
<tr>
<td>CommandManagerSelect =</td>
<td>MicroStrategy Command Manager</td>
</tr>
<tr>
<td>DeliveryEngineSelect =</td>
<td>MicroStrategy Delivery Engine</td>
</tr>
<tr>
<td>DesktopDesignerSelect =</td>
<td>MicroStrategy Desktop Designer</td>
</tr>
<tr>
<td>DesktopAnalystSelect =</td>
<td>MicroStrategy Desktop Analyst</td>
</tr>
<tr>
<td>EnterpriseManagerSelect =</td>
<td>MicroStrategy Enterprise Manager</td>
</tr>
<tr>
<td>FunctionPluginSelect =</td>
<td>MicroStrategy Function Plug-In Wizard</td>
</tr>
<tr>
<td>GISConnectorsSelect =</td>
<td>MicroStrategy GIS Connectors</td>
</tr>
<tr>
<td>IntegrityManagerSelect =</td>
<td>MicroStrategy Integrity Manager</td>
</tr>
<tr>
<td>IServerDistributionServicesSelect =</td>
<td>MicroStrategy Distribution Services</td>
</tr>
<tr>
<td>IServerOLAPServicesSelect =</td>
<td>MicroStrategy OLAP Services</td>
</tr>
<tr>
<td>IServerReportServicesSelect =</td>
<td>MicroStrategy Report Services</td>
</tr>
<tr>
<td>IServerSelect =</td>
<td>MicroStrategy Intelligence Server</td>
</tr>
<tr>
<td>IServerTransactionServicesSelect =</td>
<td>MicroStrategy Transaction Services</td>
</tr>
<tr>
<td>MobileClientSelect =</td>
<td>MicroStrategy Mobile BlackBerry client</td>
</tr>
<tr>
<td>MobileSelect =</td>
<td>MicroStrategy Mobile</td>
</tr>
<tr>
<td>MobileServerASPSelect =</td>
<td>MicroStrategy Mobile Server (ASP.NET)</td>
</tr>
<tr>
<td>MobileServerJSPSelect =</td>
<td>MicroStrategy Mobile Server (JSP)</td>
</tr>
<tr>
<td>NCSAdminSelect =</td>
<td>MicroStrategy Narrowcast Administrator</td>
</tr>
<tr>
<td>ObjectManagerSelect =</td>
<td>MicroStrategy Object Manager</td>
</tr>
</tbody>
</table>
### Options

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OfficeSelect =</td>
<td>MicroStrategy Office</td>
</tr>
<tr>
<td>PortletsSelect =</td>
<td>MicroStrategy Portlets</td>
</tr>
<tr>
<td>ProjectBuilderSelect =</td>
<td>MicroStrategy Project Builder</td>
</tr>
<tr>
<td>SDKSelect =</td>
<td>MicroStrategy SDK</td>
</tr>
<tr>
<td>SequeLinkSelect =</td>
<td>SequeLink ODBC Socket Server</td>
</tr>
<tr>
<td>ServerAdminSelect =</td>
<td>MicroStrategy Server Administrator</td>
</tr>
<tr>
<td>SubscriptionPortalSelect =</td>
<td>MicroStrategy Subscription Portal</td>
</tr>
<tr>
<td>TM1ConnectorSelect =</td>
<td>TM1 Connector for MicroStrategy</td>
</tr>
<tr>
<td>TutorialDeliveryConfigureSelect =</td>
<td>MicroStrategy Tutorial - Delivery Configuration</td>
</tr>
<tr>
<td>TutorialDeliveryInstallSelect =</td>
<td>MicroStrategy Tutorial - Delivery Installation</td>
</tr>
<tr>
<td>TutorialReportingSelect =</td>
<td>MicroStrategy Tutorial - Reporting</td>
</tr>
<tr>
<td>WebAnalystSelect =</td>
<td>MicroStrategy Web Analyst</td>
</tr>
<tr>
<td>WebMMTSelect =</td>
<td>MicroStrategy Web MMT</td>
</tr>
<tr>
<td>WebProfessionalSelect =</td>
<td>MicroStrategy Web Professional</td>
</tr>
<tr>
<td>WebReporterSelect =</td>
<td>MicroStrategy Web Reporter</td>
</tr>
<tr>
<td>WebServerASPNETSelect =</td>
<td>MicroStrategy Web Server (ASP.NET)</td>
</tr>
<tr>
<td>WebServerJSPSelect =</td>
<td>MicroStrategy Web Server JSP</td>
</tr>
<tr>
<td>WebServicesASPNETSelect =</td>
<td>MicroStrategy Web Services (ASP.NET)</td>
</tr>
<tr>
<td>WebServicesJSPSelect =</td>
<td>MicroStrategy Web Services Universal (JSP)</td>
</tr>
</tbody>
</table>

### MicroStrategy Web virtual directory

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[WebVirtualDirectory]</td>
<td>Section that specifies the virtual directory to be used for the MicroStrategy Web application.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>VirtualDirectory =</td>
<td>Enter a name for the virtual directory. The default is MicroStrategy8.</td>
</tr>
<tr>
<td>RemoveVD =</td>
<td>YES or NO. This option is for the uninstall only. Set this option to remove an existing MicroStrategy Web virtual directory from a previous installation. The default is NO.</td>
</tr>
</tbody>
</table>
## MicroStrategy Mobile Server virtual directory

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[MobileVirtualDirectory]</td>
<td>Section that specifies the virtual directory to be used for the MicroStrategy Mobile Server applications.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>VirtualDirectory =</td>
<td>Enter a name for the virtual directory. The default is MicroStrategyMobile.</td>
</tr>
<tr>
<td>ReconfigureVirtualDirectory =</td>
<td>TRUE or False. This option is relevant to upgrading MicroStrategy from a pre-9.0.1m version. TRUE replaces the virtual directory used to support MicroStrategy Mobile for BlackBerry with the new virtual directory specified for MicroStrategy Mobile Server. For more information on upgrade installations, see the Upgrade Guide.</td>
</tr>
<tr>
<td>RemoveVD =</td>
<td>YES or NO. This option is for the uninstall only. Set this option to remove an existing MicroStrategy Mobile Server virtual directory from a previous installation. The default is NO.</td>
</tr>
</tbody>
</table>

## TM1 Connector for MicroStrategy virtual directory

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TM1ConnectorVirtualDirectory]</td>
<td>Section that specifies the virtual directory to be used for the MicroStrategy Mobile Server applications.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>VirtualDirectory =</td>
<td>Enter a name for the virtual directory. The default is MicroStrategyXmla.</td>
</tr>
<tr>
<td>ReconfigureVirtualDirectory =</td>
<td>TRUE or FALSE. Set this option to TRUE if the virtual directory for the TM1 Connector for MicroStrategy should be reconfigured to support a new virtual directory.</td>
</tr>
<tr>
<td>RemoveVD =</td>
<td>YES or NO. This option is for the uninstall only. Set this option to remove an existing TM1 Connector for MicroStrategy virtual directory from a previous installation. The default is NO.</td>
</tr>
</tbody>
</table>

## Subscription Portal virtual directory

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[PortalVirtualDirectory]</td>
<td>Section that specifies the virtual directory to be used for MicroStrategy Subscription Portal.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
</tbody>
</table>
### Options

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VirtualDirectory =</td>
<td>Enter a name for the virtual directory. The default is NarrowcastServer8.</td>
</tr>
<tr>
<td>RemoveVD =</td>
<td>YES or NO. This option is for the uninstall only. Set this option to remove an existing MicroStrategy Subscription Portal virtual directory from a previous installation. The default is NO.</td>
</tr>
</tbody>
</table>

**MicroStrategy Web Services virtual directory**

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[WebServicesDirectory]</td>
<td>Section that specifies the virtual directory to be used for MicroStrategy Web Services.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>VirtualDirectory =</td>
<td>Enter a name for the virtual directory. The default is MicroStrategyWS.</td>
</tr>
<tr>
<td>RemoveVD =</td>
<td>YES or NO. This option is for the uninstall only. Set this option to remove an existing MicroStrategy Subscription Portal virtual directory from a previous installation. The default is NO.</td>
</tr>
</tbody>
</table>

**Web MMT virtual directory**

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[WebMMTVirtualDirectory]</td>
<td>Section that specifies the virtual directory to be used for MicroStrategy Web MMT.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. Set this option to TRUE to hide either the MicroStrategy Web MMT Virtual Directory Setting dialog box, or the prompt for removal of the virtual directory while MicroStrategy Web MMT is being uninstalled. The default is FALSE.</td>
</tr>
<tr>
<td>VirtualDirectory =</td>
<td>Enter a name for the virtual directory. If left empty, it takes the value of WebMMT.</td>
</tr>
<tr>
<td>ReconfigureVirtualDirectory =</td>
<td>TRUE or FALSE. Set this option to TRUE if the virtual directory for Web MMT should be reconfigured to support a new Web MMT virtual directory.</td>
</tr>
<tr>
<td>RemoveVD =</td>
<td>TRUE or FALSE. Set this option to specify whether upon uninstallation the MicroStrategy Web MMT virtual directory should be removed. The default value is FALSE.</td>
</tr>
</tbody>
</table>
Program folder

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ProgramFolder]</td>
<td>Section specifying the Windows Start name of the folder from where you can run the MicroStrategy products. The folder will be created under the default Start\Programs.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>Name =</td>
<td>Enter the name of the folder, for example, MicroStrategy.</td>
</tr>
</tbody>
</table>

Intelligence Server service account

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[IServerServiceAccount]</td>
<td>Section specifying the Windows account for the MicroStrategy Intelligence Server service. You have two options:</td>
</tr>
<tr>
<td></td>
<td>• bypass entering the account information</td>
</tr>
<tr>
<td></td>
<td>• enter the account information</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>SkipAccountSetting =</td>
<td>TRUE or FALSE. Set TRUE to bypass the service account setting in the MicroStrategy Intelligence Server Setting dialog box. If you bypass it, then the service runs with the local system account that is installing the products. The default is FALSE.</td>
</tr>
<tr>
<td>Domain =</td>
<td>Enter the domain where the account is located.</td>
</tr>
<tr>
<td>Login =</td>
<td>Enter the user name of the account to use.</td>
</tr>
<tr>
<td>Password =</td>
<td>Enter the password for the account.</td>
</tr>
<tr>
<td>ServiceStartup =</td>
<td>AUTO or MANUAL. Select to set the Intelligence Server service startup to be automatic or manual. The default is AUTO.</td>
</tr>
</tbody>
</table>

Narrowcast Server service account

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[NarrowcastServiceAccount]</td>
<td>Section specifying the Windows account from which the MicroStrategy Narrowcast Server service will run.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>SkipAccountSetting =</td>
<td>TRUE or FALSE.</td>
</tr>
<tr>
<td></td>
<td>If you specify this value as FALSE, the service account settings are not skipped and the MicroStrategy Narrowcast Server setting dialog box is displayed. Specify the details of the Windows account that the MicroStrategy Narrowcast Server services will use to log on and click Next to proceed with the installation process.</td>
</tr>
</tbody>
</table>
## MicroStrategy Web Services and Web Services Universal URL setting

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>AllowBlankURL =</td>
<td>TRUE or FALSE. Specify whether to allow a blank URL. The installation routine validates the provided URL. If no URL is provided, the user is informed that it has been left blank and needs to be configured with the MicroStrategy Office Configuration Tool. If this is set to TRUE, the user message is not displayed if the URL is left blank. The default is FALSE.</td>
</tr>
<tr>
<td>URL =</td>
<td>Enter a URL pointing to a valid MicroStrategy Web Services installation, for example, <a href="http://localhost/MicrostrategyWS/MSTRWS.asmx">http://localhost/MicrostrategyWS/MSTRWS.asmx</a></td>
</tr>
</tbody>
</table>

## MicroStrategy Office setting

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[MSOfficeLoadOptions]</td>
<td>Section specifying the options that determine if the MicroStrategy Office toolbar is loaded in the installed Microsoft Office applications.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>ConfigureExcel =</td>
<td>TRUE or FALSE. Specify to load the MicroStrategy Office toolbar by default when the Microsoft Excel application runs. This applies only if Excel is installed in the target machine. The default is TRUE.</td>
</tr>
<tr>
<td>ConfigureWord =</td>
<td>TRUE or FALSE. Specify to load the MicroStrategy Office toolbar by default when the Microsoft Word application runs. This applies only if Word is installed on the target machine. The default is TRUE.</td>
</tr>
<tr>
<td>ConfigurePowerpoint =</td>
<td>TRUE or FALSE. Specify to load the MicroStrategy Office toolbar by default when the Microsoft PowerPoint application runs. This applies only if PowerPoint is installed on the target machine. The default is TRUE.</td>
</tr>
</tbody>
</table>
Intelligence Server definition setting

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ServerDefinitionSetting]</td>
<td>Section specifying whether or not the MicroStrategy Intelligence Server will use the server definition included with the Tutorial.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>OverwriteServerDefinition =</td>
<td>TRUE or FALSE. This option relates to the Tutorial. Set this option to overwrite existing MicroStrategy Intelligence Server definitions from a previous install. The default is FALSE.</td>
</tr>
</tbody>
</table>

Tutorial reporting setting

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TutorialReportingSetting]</td>
<td>Section that specifies the DSN used to connect to the MicroStrategy Tutorial.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
<tr>
<td>OverwriteDSN =</td>
<td>TRUE or FALSE. Set this option to overwrite an existing DSN with the same name. The data source names are as follows:</td>
</tr>
<tr>
<td></td>
<td>• MicroStrategy_Tutorial_Metadata</td>
</tr>
<tr>
<td></td>
<td>• MicroStrategy_Tutorial_Data</td>
</tr>
<tr>
<td></td>
<td>The default is FALSE.</td>
</tr>
</tbody>
</table>

Start Copying Files dialog box

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Summary]</td>
<td>Section that specifies the installation summary in the Start Copying Files dialog box.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
</tbody>
</table>

MicroStrategy Installation Wizard Complete dialog box

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Finish]</td>
<td>Section that specifies the MicroStrategy Installation Wizard Complete dialog box.</td>
</tr>
<tr>
<td>HideDialog =</td>
<td>TRUE or FALSE. FALSE displays the dialog box. The default is FALSE.</td>
</tr>
</tbody>
</table>
Example of a response.ini file

[Installer]
PropertiesFilesOverwrite=FALSE
EnableTracing=FALSE
HideAllDialogs=TRUE
ForceReboot=TRUE
PreventReboot=FALSE
CheckTCPIP=TRUE
CheckIIS=TRUE
CreateShortcuts=TRUE
CheckRenameOperations=TRUE
AnalyticsOverwrite=FALSE
TutReportingOverwrite=FALSE
TutDeliveryOverwrite=FALSE
BackupFiles=FALSE
RunConfigWizard=FALSE
StopAllServices=TRUE
StopIIS=TRUE
EnableASPServices=TRUE
EnableASPNETServices=TRUE
WebMMFForWebOverwrite=FALSE
ShowWelcomeScreen=FALSE
ShowConfigWizard=FALSE
EnterpriseManagerOverwrite=FALSE
ConfigWizardResponseFile=Response.ini
LogFile=C:\Program Files\install.log

[Welcome]
HideDialog=TRUE
RemoveAll=FALSE

[UserRegistration]
HideDialog=TRUE
UserName=UserNameHere
CompanyName=CompanyNameHere
LicenseKey=CustomerLicenseKeyHere

[SetupType]
HideDialog=TRUE
Type=TYPICAL

[SuiteTarget]
HideDialog=TRUE
TargetDirectory=C:\Program Files\MicroStrategy

[ComponentSelection]
HideDialog=TRUE

### Visible Components ###
DesktopDesignerVisible=TRUE
DesktopAnalystVisible=TRUE
ArchitectVisible=TRUE
ServerAdminVisible=TRUE
ProjectBuilderVisible=TRUE
FunctionPluginVisible=FALSE
CommandManagerVisible=TRUE
EnterpriseManagerVisible=TRUE
ObjectManagerVisible=TRUE
IntegrityManagerVisible=TRUE
IServerVisible=TRUE
IServerOLAPServicesVisible=TRUE
IServerReportServicesVisible=TRUE
IServerDistributionServicesVisible=TRUE
IServerTransactionServicesVisible=TRUE
WebAnalystVisible=TRUE
WebProfessionalVisible=TRUE
WebReporterVisible=TRUE
WebServerASPNETVisible=TRUE
WebServerJSPVisible=FALSE
WebServicesASPNETVisible=TRUE
WebServicesJSPVisible=FALSE
SDKVisible=TRUE
OfficeVisible=TRUE
MobileVisible=TRUE
MobileClientVisible=TRUE
MobileServerASPVisible=TRUE
MobileServerJSPVisible=FALSE
TutorialReportingVisible=TRUE
WebMMTVisible=TRUE
AnalyticsModulesVisible=TRUE
NCAServerVisible=TRUE
DeliveryEngineVisible=TRUE
SubscriptionPortalVisible=TRUE
TutorialDeliveryInstallVisible=TRUE
TutorialDeliveryConfigureVisible=TRUE
SequeLinkVisible=TRUE
PortletsVisible=TRUE
TM1ConnectorVisible=TRUE
GISConnectorsVisible=TRUE
### Components To Install (TRUE) or Remove (FALSE) ###
DesktopDesignerSelect=TRUE
DesktopAnalystSelect=TRUE
ArchitectSelect=TRUE
ServerAdminSelect=TRUE
ProjectBuilderSelect=TRUE
FunctionPluginSelect=FALSE
CommandManagerSelect=TRUE
EnterpriseManagerSelect=TRUE
ObjectManagerSelect=TRUE
IntegrityManagerSelect=TRUE
IServerSelect=TRUE
IServerOLAPServicesSelect=TRUE
IServerReportServicesSelect=TRUE
IServerDistributionServicesSelect=TRUE
IServerTransactionServicesSelect=TRUE
WebAnalystSelect=TRUE
WebProfessionalSelect=TRUE
WebReporterSelect=TRUE
WebServerASPNETSelect=TRUE
WebServerJSPSelect=FALSE
WebServicesASPNETSelect=TRUE
WebServicesJSPSelect=FALSE
SDKSelect=TRUE
OfficeSelect=TRUE
MobileSelect=TRUE
MobileClientSelect=TRUE
MobileServerASPSelect=TRUE
MobileServerJSPSelect=FALSE
TutorialReportingSelect=TRUE
WebMMTSelect=TRUE
AnalyticsModulesSelect=TRUE
NCSAdminSelect=TRUE
DeliveryEngineSelect=TRUE
SubscriptionPortalSelect=TRUE
TutorialDeliveryInstallSelect=TRUE
TutorialDeliveryConfigureSelect=TRUE
SequeLinkSelect=TRUE
PortletsSelect=TRUE
TM1ConnectorSelect=TRUE
GISConnectorsSelect=TRUE

[ServerDefinitionSetting]
HideDialog=TRUE
OverwriteServerDefinition=FALSE

[TutorialReportingSetting]
HideDialog=TRUE
OverwriteDSN=FALSE

[WebVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=MicroStrategy
ReconfigureVirtualDirectory=TRUE

[MobileVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=MicroStrategyMobile
ReconfigureVirtualDirectory=TRUE

[PortalVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=NarrowcastServer
ReconfigureVirtualDirectory=TRUE

[WebServicesVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=MicroStrategyWS
ReconfigureVirtualDirectory=TRUE

[WebMMTVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=WebMMT
ReconfigureVirtualDirectory=TRUE

[OfficeWebServicesURL]
HideDialog=TRUE
AllowBlankURL=FALSE
URL=http://localhost/MicroStrategyWS/mstrws.asmx

[MSOfficeLoadOptions]
HideDialog=TRUE
ConfigureExcel=TRUE
ConfigurePowerpoint=TRUE
ConfigureWord=TRUE

[IServerServiceAccount]
HideDialog=TRUE
SkipAccountSetting=FALSE
Login=NT_UserLoginHere
Password=UserPasswordHere  
Domain=DomainHere  
ServiceStartUp=AUTO

[NarrowcastServiceAccount]  
HideDialog=TRUE  
SkipAccountSetting=FALSE  
Login=NT_UserLoginHere  
Password=UserPasswordHere  
Domain=DomainHere

[Programfolder]  
HideDialog=TRUE  
Name=MicroStrategy

[Summary]  
HideDialog=TRUE

[Finish]  
HideDialog=TRUE

Copy and paste this example to create a `response.ini` file. Replace any text between angled brackets (`<>`) with your own specific information. For example, change `UserName=UserNamerHere` to `UserName=jsmith`. Make sure you check for correct spaces and new lines in all file paths.

⚠️ Your license key determines which MicroStrategy components will be available for your installation. For example, if your license key does not include MicroStrategy OLAP Services, then you cannot use `IServerOLAPServicesSelect=TRUE` and `IServerOLAPServicesVisible=TRUE` to install these components.

**Example of a `response.ini` file: Installs Intelligence Server, SDK, Enterprise Manager, and other components**

[Installer]  
HideAllDialogs=TRUE  
ForceReboot=TRUE  
StopAllServices=TRUE  
StopIIS=TRUE  
EnterpriseManagerOverwrite=TRUE

[Welcome]  
HideDialog=TRUE  
RemoveAll=FALSE
[UserRegistration]
HideDialog=TRUE
UserName=UserNameHere
CompanyName=CompanyNameHere
LicenseKey=CustomerLicenseKeyHere

[SetupType]
HideDialog=TRUE
Type=TYPICAL

[SuiteTarget]
HideDialog=TRUE
TargetDirectory=C:\Program Files\MicroStrategy

[ComponentSelection]
HideDialog=TRUE

### Visible Components ###
ServerAdminVisible=TRUE
CommandManagerVisible=TRUE
EnterpriseManagerVisible=TRUE
ObjectManagerVisible=TRUE
IServerVisible=TRUE
SDKVisible=TRUE
IServerOLAPServicesVisible=TRUE
IServerReportServicesVisible=TRUE
IServerDistributionServicesVisible=TRUE
IServerTransactionServicesVisible=TRUE
IntegrityManagerVisible=TRUE
DesktopDesignerVisible=FALSE
DesktopAnalystVisible=FALSE
ArchitectVisible=FALSE
ProjectBuilderVisible=FALSE
FunctionPluginVisible=FALSE
WebAnalystVisible=FALSE
WebProfessionalVisible=FALSE
WebReporterVisible=FALSE
WebServerASPNETVisible=FALSE
WebServerJSPVisible=FALSE
WebServicesASPNETVisible=FALSE
WebServicesJSPVisible=FALSE
OfficeVisible=FALSE
MobileVisible=FALSE
MobileClientVisible=FALSE
MobileServerASPVisible=FALSE
MobileServerJSPVisible=FALSE
TutorialReportingVisible=FALSE
WebMMTVisible=FALSE
AnalyticsModulesVisible=FALSE
NCSAdminVisible=FALSE
DeliveryEngineVisible=FALSE
SubscriptionPortalVisible=FALSE
TutorialDeliveryInstallVisible=FALSE
TutorialDeliveryConfigureVisible=FALSE
SequeLinkVisible=FALSE
PortletsVisible=FALSE
TM1ConnectorVisible=FALSE
GISConnectorsVisible=FALSE

### Components To Install (TRUE) or Remove (FALSE) ###
ServerAdminSelect=TRUE
CommandManagerSelect=TRUE
EnterpriseManagerSelect=TRUE
ObjectManagerSelect=TRUE
IServerSelect=TRUE
SDKSelect=TRUE
IServerOLAPServicesSelect=TRUE
IServerReportServicesSelect=TRUE
IServerDistributionServicesSelect=TRUE
IServerTransactionServicesSelect=TRUE
IntegrityManagerSelect=TRUE
DesktopDesignerSelect=FALSE
DesktopAnalystSelect=FALSE
ArchitectSelect=FALSE
ProjectBuilderSelect=FALSE
FunctionPluginSelect=FALSE
WebAnalystSelect=FALSE
WebProfessionalSelect=FALSE
WebReporterSelect=FALSE
WebServerASPNETSelect=FALSE
WebServerJSPSelect=FALSE
WebServicesASPNETSelect=FALSE
WebServicesJSPSelect=FALSE
OfficeSelect=FALSE
MobileSelect=FALSE
MobileClientSelect=FALSE
MobileServerASPSelect=FALSE
MobileServerJSPSelect=FALSE
TutorialReportingSelect=FALSE
WebMMTSelect=FALSE
AnalyticsModulesSelect=FALSE
NCSAdminSelect=FALSE
DeliveryEngineSelect=FALSE
SubscriptionPortalSelect=FALSE
TutorialDeliveryInstallSelect=FALSE
TutorialDeliveryConfigureSelect=FALSE
SequeLinkSelect=FALSE
PortletsSelect=FALSE
TM1ConnectorSelect=FALSE
GISConnectorsSelect=FALSE

-IServerServiceAccount-
HideDialog=TRUE
SkipAccountSetting=FALSE
Login=UID
Password=PWD
Domain=DOMAIN
ServiceStartUp=AUTO

-[Programfolder]-
HideDialog=TRUE
Name=MicroStrategy

-[Summary]-
HideDialog=TRUE

-[Finish]-
HideDialog=TRUE

Copy and paste this example to create a response.ini file. Replace any text between angled brackets (<> ) with your own specific information. For example, change UserName=UserNameHere to UserName=jsmith. Make sure to check that all file paths are entered with correct spacing.

Your license key determines which MicroStrategy components will be available for your installation. For example, if your license key does not include MicroStrategy OLAP Services, then you cannot use IServerOLAPServicesSelect=TRUE and IServerOLAPServicesVisible=TRUE to install these components.

**Example of a response.ini file: Installs Intelligence Server components**

[Installer]
HideAllDialogs=TRUE
ForceReboot=TRUE
StopAllServices=TRUE

[Welcome]
HideDialog=TRUE
RemoveAll=FALSE

[UserRegistration]
HideDialog=TRUE
UserName=UserNameHere
CompanyName=CompanyNameHere
LicenseKey=CustomerLicenseKeyHere

[SetupType]
HideDialog=TRUE
Type=TYPICAL

[SuiteTarget]
HideDialog=TRUE
TargetDirectory=C:\Program Files\MicroStrategy

[ComponentSelection]
HideDialog=TRUE

### Visible Components ###
ServerAdminVisible=TRUE
IServerVisible=TRUE
IServerOLAPServicesVisible=TRUE
IServerReportServicesVisible=TRUE
IServerDistributionServicesVisible=TRUE
IServerTransactionServicesVisible=TRUE
CommandManagerVisible=FALSE
EnterpriseManagerVisible=FALSE
ObjectManagerVisible=FALSE
SDKVisible=FALSE
IntegrityManagerVisible=FALSE
DesktopDesignerVisible=FALSE
DesktopAnalystVisible=FALSE
ArchitectVisible=FALSE
ProjectBuilderVisible=FALSE
FunctionPluginVisible=FALSE
WebAnalystVisible=FALSE
WebProfessionalVisible=FALSE
WebReporterVisible=FALSE
WebServerASPNETVisible=FALSE
WebServerJSPVisible=FALSE
WebServicesASPNETVisible=FALSE
WebServicesJSPVisible=FALSE
OfficeVisible=FALSE
MobileVisible=FALSE
MobileClientVisible=FALSE
MobileServerASPVisible=FALSE
MobileServerJSPVisible=FALSE
TutorialReportingVisible=FALSE
WebMMTVisible=FALSE
AnalyticsModulesVisible=FALSE
NCSAdminVisible=FALSE
DeliveryEngineVisible=FALSE
SubscriptionPortalVisible=FALSE
TutorialDeliveryInstallVisible=FALSE
TutorialDeliveryConfigureVisible=FALSE
SequeLinkVisible=FALSE
PortletsVisible=FALSE
TM1ConnectorVisible=FALSE
GISConnectorsVisible=FALSE

### Components To Install (TRUE) or Remove (FALSE) ###
ServerAdminSelect=TRUE
IServerSelect=TRUE
IServerOLAPServicesSelect=TRUE
IServerReportServicesSelect=TRUE
IServerDistributionServicesSelect=TRUE
IServerTransactionServicesSelect=TRUE
CommandManagerSelect=FALSE
EnterpriseManagerSelect=FALSE
ObjectManagerSelect=FALSE
SDKSelect=FALSE
IntegrityManagerSelect=FALSE
DesktopDesignerSelect=FALSE
DesktopAnalystSelect=FALSE
ArchitectSelect=FALSE
ProjectBuilderSelect=FALSE
FunctionPluginSelect=FALSE
WebAnalystSelect=FALSE
WebProfessionalSelect=FALSE
WebReporterSelect=FALSE
WebServerASPNETSelect=FALSE
WebServerJSPSelect=FALSE
WebServicesASPNETSelect=FALSE
WebServicesJSPSelect=FALSE
OfficeSelect=FALSE
MobileSelect=FALSE
MobileClientSelect=FALSE
MobileServerASPSel=FALSE
MobileServerJSPSelect=FALSE
TutorialReportingSelect=FALSE
WebMMTSelect=FALSE
AnalyticsModulesSelect=FALSE
NCAdminSelect=FALSE
DeliveryEngineSelect=FALSE
SubscriptionPortalSelect=FALSE
TutorialDeliveryInstallSelect=FALSE
TutorialDeliveryConfigureSelect=FALSE
SequeLinkSelect=FALSE
PortletsSelect=FALSE
TM1ConnectorSelect=FALSE
GISConnectorsSelect=FALSE

[IServerServiceAccount]
HideDialog=TRUE
SkipAccountSetting=FALSE
Login=UID
Password=PWD
Domain=DOMAIN
ServiceStartUp=AUTO

[Programfolder]
HideDialog=TRUE
Name=MicroStrategy

[Summary]
HideDialog=TRUE

[Finish]
HideDialog=TRUE

Copy and paste this example to create a response.ini file. Replace any text between angled brackets (<>) with your own specific information. For example, change `UserName=UserNameHere` to `UserName=jsmith`. Make sure to check that all file paths are entered with correct spacing.

Your license key determines which MicroStrategy components will be available for your installation. For example, if your license key does not include MicroStrategy OLAP Services, then you cannot use `IServerOLAPServicesSelect=TRUE` and...
IServerOLAPServicesVisible=TRUE to install these components.

Example of a response.ini file: Installs MicroStrategy Web components

[Installer]
HideAllDialogs=TRUE
ForceReboot=TRUE
StopAllServices=TRUE
StopIIS=TRUE
EnableASPServices=TRUE
EnableASPNETServices=TRUE

[Welcome]
HideDialog=TRUE
RemoveAll=FALSE

[UserRegistration]
HideDialog=TRUE
UserName=UserNameHere
CompanyName=CompanyNameHere
LicenseKey=CustomerLicenseKeyHere

[SetupType]
HideDialog=TRUE
Type=TYPICAL

[SuiteTarget]
HideDialog=TRUE
TargetDirectory=C:\Program Files\MicroStrategy

[ComponentSelection]
HideDialog=TRUE

### Visible Components ###
WebReporterVisible=TRUE
WebAnalystVisible=TRUE
WebProfessionalVisible=TRUE
WebServerASPNETVisible=TRUE
WebServerJSPVisible=TRUE
WebServicesASPNETVisible=TRUE
WebServicesJSPVisible=TRUE
ServerAdminVisible=FALSE
IServerVisible=FALSE
IServerOLAPServicesVisible=FALSE
IServerReportServicesVisible=FALSE
IServerDistributionServicesVisible=FALSE
IServerTransactionServicesVisible=FALSE
CommandManagerVisible=FALSE
EnterpriseManagerVisible=FALSE
ObjectManagerVisible=FALSE
SDKVisible=FALSE
IntegrityManagerVisible=FALSE
DesktopDesignerVisible=FALSE
DesktopAnalystVisible=FALSE
ArchitectVisible=FALSE
ProjectBuilderVisible=FALSE
FunctionPluginVisible=FALSE
OfficeVisible=FALSE
MobileVisible=FALSE
MobileClientVisible=FALSE
MobileServerASPVisible=FALSE
MobileServerJSPVisible=FALSE
TutorialReportingVisible=FALSE
WebMMTVisible=FALSE
AnalyticsModulesVisible=FALSE
NCSAdminVisible=FALSE
DeliveryEngineVisible=FALSE
SubscriptionPortalVisible=FALSE
TutorialDeliveryInstallVisible=FALSE
TutorialDeliveryConfigureVisible=FALSE
SequeLinkVisible=FALSE
PortletsVisible=FALSE
TM1ConnectorVisible=FALSE
GISConnectorsVisible=FALSE

### Components To Install (TRUE) or Remove (FALSE) ###
WebReporterSelect=TRUE
WebAnalystSelect=TRUE
WebProfessionalSelect=TRUE
WebServerASPNETSelect=TRUE
WebServerJSPSelect=TRUE
WebServicesASPNETSelect=TRUE
WebServicesJSPSelect=TRUE
ServerAdminSelect=FALSE
IServerSelect=FALSE
IServerOLAPServicesSelect=FALSE
IServerReportServicesSelect=FALSE
IServerDistributionServicesSelect=FALSE
IServerTransactionServicesSelect=FALSE
CommandManagerSelect=FALSE
EnterpriseManagerSelect=FALSE
ObjectManagerSelect=FALSE
SDKSelect=FALSE
IntegrityManagerSelect=FALSE
DesktopDesignerSelect=FALSE
DesktopAnalystSelect=FALSE
ArchitectSelect=FALSE
ProjectBuilderSelect=FALSE
FunctionPluginSelect=FALSE
OfficeSelect=FALSE
MobileSelect=FALSE
MobileClientSelect=FALSE
MobileServerASPSeselect=FALSE
MobileServerJSPSelect=FALSE
TutorialReportingSelect=FALSE
WebMMTSelect=FALSE
AnalyticsModulesSelect=FALSE
NCSAdminSelect=FALSE
DeliveryEngineSelect=FALSE
SubscriptionPortalSelect=FALSE
TutorialDeliveryInstallSelect=FALSE
TutorialDeliveryConfigureSelect=FALSE
SequeLinkSelect=FALSE
PortletsSelect=FALSE
TM1ConnectorSelect=FALSE
GISConnectorsSelect=FALSE

[WebVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=MicroStrategy
ReconfigureVirtualDirectory=TRUE

[WebServicesVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=MicroStrategyWS
ReconfigureVirtualDirectory=TRUE

[Programfolder]
HideDialog=TRUE
Name=MicroStrategy

[Summary]
HideDialog=TRUE
Copy and paste this example to create a response.ini file. Replace any text between angled brackets (<>) with your own specific information. For example, change UserName=UserNameHere to UserName=jsmith. Make sure to check that all file paths are entered with correct spacing.

Your license key determines which MicroStrategy components will be available for your installation. For example, if your license key does not include MicroStrategy Web Reporter, then you cannot use WebReporterVisible=TRUE and WebReporterSelect=TRUE to install these components.

Example of a response.ini file: Installs SDK components

[Installer]
HideAllDialogs=TRUE
ForceReboot=TRUE
StopAllServices=TRUE
StopIIS=TRUE

[Welcome]
HideDialog=TRUE
RemoveAll=FALSE

[UserRegistration]
HideDialog=TRUE
UserName=UserNameHere
CompanyName=CompanyNameHere
LicenseKey=CustomerLicenseKeyHere

[SetupType]
HideDialog=TRUE
Type=TYPICAL

[SuiteTarget]
HideDialog=TRUE
TargetDirectory=C:\Program Files\MicroStrategy

[ComponentSelection]
HideDialog=TRUE

### Visible Components ###
SDKVisible=TRUE
ServerAdminVisible=FALSE
CommandManagerVisible=FALSE
EnterpriseManagerVisible=FALSE
ObjectManagerVisible=FALSE
IServerVisible=FALSE
IServerOLAPServicesVisible=FALSE
IServerReportServicesVisible=FALSE
IServerDistributionServicesVisible=FALSE
IServerTransactionServicesVisible=FALSE
IntegrityManagerVisible=FALSE
DesktopDesignerVisible=FALSE
DesktopAnalystVisible=FALSE
ArchitectVisible=FALSE
ProjectBuilderVisible=FALSE
FunctionPluginVisible=FALSE
WebAnalystVisible=FALSE
WebProfessionalVisible=FALSE
WebReporterVisible=FALSE
WebServerASPNETVisible=FALSE
WebServerJSPVisible=FALSE
WebServicesASPNETVisible=FALSE
WebServicesJSPVisible=FALSE
OfficeVisible=FALSE
MobileVisible=FALSE
MobileClientVisible=FALSE
MobileServerASPVisible=FALSE
MobileServerJSPVisible=FALSE
TutorialReportingVisible=FALSE
WebMMTVisible=FALSE
AnalyticsModulesVisible=FALSE
NCSAdminVisible=FALSE
DeliveryEngineVisible=FALSE
SubscriptionPortalVisible=FALSE
TutorialDeliveryInstallVisible=FALSE
TutorialDeliveryConfigureVisible=FALSE
SequeLinkVisible=FALSE
PortletsVisible=FALSE
TM1ConnectorVisible=FALSE
GISConnectorsVisible=FALSE

### Components To Install (TRUE) or Remove (FALSE) ###
SDKSelect=TRUE
ServerAdminSelect=FALSE
CommandManagerSelect=FALSE
EnterpriseManagerSelect=FALSE
ObjectManagerSelect=FALSE
IServerSelect=FALSE
IServerOLAPServicesSelect=FALSE
IServerReportServicesSelect=FALSE
IServerDistributionServicesSelect=FALSE
IServerTransactionServicesSelect=FALSE
IntegrityManagerSelect=FALSE
DesktopDesignerSelect=FALSE
DesktopAnalystSelect=FALSE
ArchitectSelect=FALSE
ProjectBuilderSelect=FALSE
FunctionPluginSelect=FALSE
WebAnalystSelect=FALSE
WebProfessionalSelect=FALSE
WebReporterSelect=FALSE
WebServerASPNETSelect=FALSE
WebServerJSPSelect=FALSE
WebServicesASPNETSelect=FALSE
WebServicesJSPSelect=FALSE
OfficeSelect=FALSE
MobileSelect=FALSE
MobileClientSelect=FALSE
MobileServerASPSelect=FALSE
MobileServerJSPSelect=FALSE
TutorialReportingSelect=FALSE
WebMMTSelect=FALSE
AnalyticsModulesSelect=FALSE
NCSAdminSelect=FALSE
DeliveryEngineSelect=FALSE
SubscriptionPortalSelect=FALSE
TutorialDeliveryInstallSelect=FALSE
TutorialDeliveryConfigureSelect=FALSE
SequeLinkSelect=FALSE
PortletsSelect=FALSE
TM1ConnectorSelect=FALSE
GISConnectorsSelect=FALSE

[Programfolder]
HideDialog=TRUE
Name=MicroStrategy

[Summary]
HideDialog=TRUE

[Finish]
HideDialog=TRUE

Copy and paste this example to create a response.ini file. Replace any text between angled brackets (<>), with your own specific information. For example, change UserName=UserNameHere to UserName=jsmith. Make sure to check that all file paths are entered with correct spacing.

Your license key determines which MicroStrategy components will be available for your installation. For example, if your license key does not include MicroStrategy SDK, then you cannot use SDKVisible=TRUE and SDKSelect=TRUE to install these components.

Example of a response.ini file: Installs Enterprise Manager components

[Installer]
HideAllDialogs=TRUE
ForceReboot=TRUE
StopAllServices=TRUE
EnterpriseManagerOverwrite=TRUE

[Welcome]
HideDialog=TRUE
RemoveAll=FALSE

[UserRegistration]
HideDialog=TRUE
UserName=UserNameHere
CompanyName=CompanyNameHere
LicenseKey=CustomerLicenseKeyHere

[SetupType]
HideDialog=TRUE
Type=TYPICAL

[SuiteTarget]
HideDialog=TRUE
TargetDirectory=C:\Program Files\MicroStrategy

[ComponentSelection]
HideDialog=TRUE

### Visible Components ###
EnterpriseManagerVisible=TRUE
SDKVisible=FALSE
ServerAdminVisible=FALSE
CommandManagerVisible=FALSE
ObjectManagerVisible=FALSE
IServerVisible=FALSE
IServerOLAPServicesVisible=FALSE
IServerReportServicesVisible=FALSE
IServerDistributionServicesVisible=FALSE
IServerTransactionServicesVisible=FALSE
IntegrityManagerVisible=FALSE
DesktopDesignerVisible=FALSE
DesktopAnalystVisible=FALSE
ArchitectVisible=FALSE
ProjectBuilderVisible=FALSE
FunctionPluginVisible=FALSE
WebAnalystVisible=FALSE
WebProfessionalVisible=FALSE
WebReporterVisible=FALSE
WebServerASPNETVisible=FALSE
WebServerJSPVisible=FALSE
WebServicesASPNETVisible=FALSE
WebServicesJSPVisible=FALSE
OfficeVisible=FALSE
MobileVisible=FALSE
MobileClientVisible=FALSE
MobileServerASPVisible=FALSE
MobileServerJSPVisible=FALSE
TutorialReportingVisible=FALSE
WebMMTVisible=FALSE
AnalyticsModulesVisible=FALSE
NCSAdminVisible=FALSE
DeliveryEngineVisible=FALSE
SubscriptionPortalVisible=FALSE
TutorialDeliveryInstallVisible=FALSE
TutorialDeliveryConfigureVisible=FALSE
SequeLinkVisible=FALSE
PortletsVisible=FALSE
TM1ConnectorVisible=FALSE
GISConnectorsVisible=FALSE

### Components To Install (TRUE) or Remove (FALSE) ###
EnterpriseManagerSelect=TRUE
SDKSelect=FALSE
ServerAdminSelect=FALSE
CommandManagerSelect=FALSE
ObjectManagerSelect=FALSE
IServerSelect=FALSE
IServerOLAPServicesSelect=FALSE
IServerReportServicesSelect=FALSE
IServerDistributionServicesSelect=FALSE
IServerTransactionServicesSelect=FALSE
IntegrityManagerSelect=FALSE
DesktopDesignerSelect=FALSE
DestopAnalystSelect=FALSE
ArchitectSelect=FALSE
ProjectBuilderSelect=FALSE
FunctionPluginSelect=FALSE
WebAnalystSelect=FALSE
WebProfessionalSelect=FALSE
WebReporterSelect=FALSE
WebServerASPNETSelect=FALSE
WebServerJSPSelect=FALSE
WebServicesASPNETSelect=FALSE
WebServicesJSPSelect=FALSE
OfficeSelect=FALSE
MobileSelect=FALSE
MobileClientSelect=FALSE
MobileServerASPSelect=FALSE
MobileServerJSPSelect=FALSE
TutorialReportingSelect=FALSE
WebMMTSelect=FALSE
AnalyticsModulesSelect=FALSE
NCSAdminSelect=FALSE
DeliveryEngineSelect=FALSE
SubscriptionPortalSelect=FALSE
TutorialDeliveryInstallSelect=FALSE
TutorialDeliveryConfigureSelect=FALSE
SequeLinkSelect=FALSE
PortletsSelect=FALSE
TM1ConnectorSelect=FALSE
GISConnectorsSelect=FALSE

[Programfolder]
HideDialog=TRUE
Name=MicroStrategy

[Summary]
HideDialog=TRUE

[Finish]
HideDialog=TRUE
Copy and paste this example to create a response.ini file. Replace any text between angled brackets (<>), with your own specific information. For example, change `UserName=UserNameHere` to `UserName=jsmith`. Make sure to check that all file paths are entered with correct spacing.

Your license key determines which MicroStrategy components will be available for your installation. For example, if your license key does not include MicroStrategy Enterprise Manager, then you cannot use `EnterpriseManagerVisible=TRUE` and `EnterpriseManagerSelect=TRUE` to install these components.

**Example of a response.ini file: Installs Desktop, Architect, and other components**

```ini
[Installer]
HideAllDialogs=TRUE
ForceReboot=TRUE
StopAllServices=TRUE
EnterpriseManagerOverwrite=TRUE

[Welcome]
HideDialog=TRUE
RemoveAll=FALSE

[UserRegistration]
HideDialog=TRUE
UserName=UserNameHere
CompanyName=CompanyNameHere
LicenseKey=CustomerLicenseKeyHere

[SetupType]
HideDialog=TRUE
Type=TYPICAL

[SuiteTarget]
HideDialog=TRUE
TargetDirectory=C:\Program Files\MicroStrategy

[ComponentSelection]
HideDialog=TRUE

### Visible Components ###
DesktopAnalystVisible=TRUE
DesktopDesignerVisible=TRUE
ArchitectVisible=TRUE
```
ProjectBuilderVisible=TRUE
CommandManagerVisible=TRUE
EnterpriseManagerVisible=TRUE
ObjectManagerVisible=TRUE
FunctionPluginVisible=TRUE
SDKVisible=FALSE
ServerAdminVisible=FALSE
IServerVisible=FALSE
IServerOLAPServicesVisible=FALSE
IServerReportServicesVisible=FALSE
IServerDistributionServicesVisible=FALSE
IServerTransactionServicesVisible=FALSE
IntegrityManagerVisible=FALSE
WebAnalystVisible=FALSE
WebProfessionalVisible=FALSE
WebReporterVisible=FALSE
WebServerASPNETVisible=FALSE
WebServerJSPVisible=FALSE
WebServicesASPNETVisible=FALSE
WebServicesJSPVisible=FALSE
OfficeVisible=FALSE
MobileVisible=FALSE
MobileClientVisible=FALSE
MobileServerASPVisible=FALSE
MobileServerJSPVisible=FALSE
TutorialReportingVisible=FALSE
WebMMTVisible=FALSE
AnalyticsModulesVisible=FALSE
NCSAdminVisible=FALSE
DeliveryEngineVisible=FALSE
SubscriptionPortalVisible=FALSE
TutorialDeliveryInstallVisible=FALSE
TutorialDeliveryConfigureVisible=FALSE
SequeLinkVisible=FALSE
PortletsVisible=FALSE
TM1ConnectorVisible=FALSE
GISConnectorsVisible=FALSE

### Components To Install (TRUE) or Remove (FALSE) ###
DesktopAnalystSelect=TRUE
DesktopDesignerSelect=TRUE
ArchitectSelect=TRUE
ProjectBuilderSelect=TRUE
CommandManagerSelect=TRUE
EnterpriseManagerSelect=TRUE
ObjectManagerSelect=TRUE
FunctionPluginSelect=TRUE
SDKSelect=FALSE
ServerAdminSelect=FALSE
IServerSelect=FALSE
IServerOLAPServicesSelect=FALSE
IServerReportServicesSelect=FALSE
IServerDistributionServicesSelect=FALSE
IServerTransactionServicesSelect=FALSE
IntegrityManagerSelect=FALSE
WebAnalystSelect=FALSE
WebProfessionalSelect=FALSE
WebReporterSelect=FALSE
WebServerASPNETSelect=FALSE
WebServerJSPSelect=FALSE
WebServicesASPNETSelect=FALSE
WebServicesJSPSelect=FALSE
OfficeSelect=FALSE
MobileSelect=FALSE
MobileClientSelect=FALSE
MobileServerASPSelect=FALSE
MobileServerJSPSelect=FALSE
TutorialReportingSelect=FALSE
WebMMTSelect=FALSE
AnalyticsModulesSelect=FALSE
NCSAdminSelect=FALSE
DeliveryEngineSelect=FALSE
SubscriptionPortalSelect=FALSE
TutorialDeliveryInstallSelect=FALSE
TutorialDeliveryConfigureSelect=FALSE
SequeLinkSelect=FALSE
PortletsSelect=FALSE
TM1ConnectorSelect=FALSE
GISConnectorsSelect=FALSE

[Programfolder]
HideDialog=TRUE
Name=MicroStrategy

[Summary]
HideDialog=TRUE

[Finish]
HideDialog=TRUE
Copy and paste this example to create a response.ini file. Replace any text between angled brackets (<> ) with your own specific information. Make sure to check that all file paths are entered with correct spacing.

Your license key determines which MicroStrategy components will be available for your installation. For example, if your license key does not include MicroStrategy Desktop Designer, then you cannot use DesktopDesignerVisible=TRUE and DesktopDesignerSelect=TRUE to install these components.

Example of a response.ini file: Installs Desktop and Architect components

[Installer]
HideAllDialogs=TRUE
ForceReboot=TRUE
StopAllServices=TRUE

[Welcome]
HideDialog=TRUE
RemoveAll=FALSE

[UserRegistration]
HideDialog=TRUE
UserName=UserNameHere
CompanyName=CompanyNameHere
LicenseKey=CustomerLicenseKeyHere

[SetupType]
HideDialog=TRUE
Type=Typical

[SuiteTarget]
HideDialog=TRUE
TargetDirectory=C:\Program Files\MicroStrategy

[ComponentSelection]
HideDialog=TRUE

### Visible Components ###
DesktopAnalystVisible=TRUE
DesktopDesignerVisible=TRUE
ArchitectVisible=TRUE
ProjectBuilderVisible=TRUE
FunctionPluginVisible=TRUE
CommandManagerVisible=FALSE
EnterpriseManagerVisible=FALSE
ObjectManagerVisible=FALSE
SDKVisible=FALSE
ServerAdminVisible=FALSE
IServerVisible=FALSE
IServerOLAPServicesVisible=FALSE
IServerReportServicesVisible=FALSE
IServerDistributionServicesVisible=FALSE
IServerTransactionServicesVisible=FALSE
IntegrityManagerVisible=FALSE
WebAnalystVisible=FALSE
WebProfessionalVisible=FALSE
WebReporterVisible=FALSE
WebServerASPNETVisible=FALSE
WebServerJSPVisible=FALSE
WebServicesASPNETVisible=FALSE
WebServicesJSPVisible=FALSE
OfficeVisible=FALSE
MobileVisible=FALSE
MobileClientVisible=FALSE
MobileServerASPVisible=FALSE
MobileServerJSPVisible=FALSE
TutorialReportingVisible=FALSE
WebMMTVisible=FALSE
AnalyticsModulesVisible=FALSE
NCSAdminVisible=FALSE
DeliveryEngineVisible=FALSE
SubscriptionPortalVisible=FALSE
TutorialDeliveryInstallVisible=FALSE
TutorialDeliveryConfigureVisible=FALSE
SequeLinkVisible=FALSE
PortletsVisible=FALSE
TM1ConnectorVisible=FALSE
GISConnectorsVisible=FALSE

### Components To Install (TRUE) or Remove (FALSE) ###
DesktopAnalystSelect=TRUE
DesktopDesignerSelect=TRUE
ArchitectSelect=TRUE
ProjectBuilderSelect=TRUE
FunctionPluginSelect=TRUE
CommandManagerSelect=FALSE
EnterpriseManagerSelect=FALSE
ObjectManagerSelect=FALSE
[Programfolder]
HideDialog=TRUE
Name=MicroStrategy

[Summary]
HideDialog=TRUE

[Finish]
HideDialog=TRUE

SDKSelect=FALSE
ServerAdminSelect=FALSE
IServerSelect=FALSE
IServerOLAPServicesSelect=FALSE
IServerReportServicesSelect=FALSE
IServerDistributionServicesSelect=FALSE
IServerTransactionServicesSelect=FALSE
IntegrityManagerSelect=FALSE
WebAnalystSelect=FALSE
WebProfessionalSelect=FALSE
WebReporterSelect=FALSE
WebServerASPNETSelect=FALSE
WebServerJSPSelect=FALSE
WebServicesASPNETSelect=FALSE
WebServicesJSPSelect=FALSE
OfficeSelect=FALSE
MobileSelect=FALSE
MobileClientSelect=FALSE
MobileServerASPSelect=FALSE
MobileServerJSPSelect=FALSE
TutorialReportingSelect=FALSE
WebMMTSelect=FALSE
AnalyticsModulesSelect=FALSE
NCSSAdminSelect=FALSE
DeliveryEngineSelect=FALSE
SubscriptionPortalSelect=FALSE
TutorialDeliveryInstallSelect=FALSE
TutorialDeliveryConfigureSelect=FALSE
SequeLinkSelect=FALSE
PortletsSelect=FALSE
TM1ConnectorSelect=FALSE
GISConnectorsSelect=FALSE

[Programfolder]
HideDialog=TRUE
Name=MicroStrategy

[Summary]
HideDialog=TRUE

[Finish]
HideDialog=TRUE
Copy and paste this example to create a response.ini file. Replace any text between angled brackets (<>) with your own specific information. Make sure to check that all file paths are entered with correct spacing.

Your license key determines which MicroStrategy components will be available for your installation. For example, if your license key does not include MicroStrategy Desktop Designer, then you cannot use DesktopDesignerVisible=TRUE and DesktopDesignerSelect=TRUE to install these components.

**Example of a response.ini file: Installs Desktop components**

```
[Installer]
HideAllDialogs=TRUE
ForceReboot=TRUE
StopAllServices=TRUE

[Welcome]
HideDialog=TRUE
RemoveAll=FALSE

[UserRegistration]
HideDialog=TRUE
UserName=UserNameHere
CompanyName=CompanyNameHere
LicenseKey=CustomerLicenseKeyHere

[SetupType]
HideDialog=TRUE
Type=TYPICAL

[SuiteTarget]
HideDialog=TRUE
TargetDirectory=C:\Program Files\MicroStrategy

[ComponentSelection]
HideDialog=TRUE

### Visible Components ###
DesktopAnalystVisible=TRUE
DesktopDesignerVisible=TRUE
ArchitectVisible=FALSE
ProjectBuilderVisible=FALSE
FunctionPluginVisible=FALSE
```
CommandManagerVisible=FALSE
EnterpriseManagerVisible=FALSE
ObjectManagerVisible=FALSE
SDKVisible=FALSE
ServerAdminVisible=FALSE
IServerVisible=FALSE
IServerOLAPServicesVisible=FALSE
IServerReportServicesVisible=FALSE
IServerDistributionServicesVisible=FALSE
IServerTransactionServicesVisible=FALSE
IntegrityManagerVisible=FALSE
WebAnalystVisible=FALSE
WebProfessionalVisible=FALSE
WebReporterVisible=FALSE
WebServerASPNETVisible=FALSE
WebServerJSPVisible=FALSE
WebServicesASPNETVisible=FALSE
WebServicesJSPVisible=FALSE
OfficeVisible=FALSE
MobileVisible=FALSE
MobileClientVisible=FALSE
MobileServerASPVisible=FALSE
MobileServerJSPVisible=FALSE
TutorialReportingVisible=FALSE
WebMMTVisible=FALSE
AnalyticsModulesVisible=FALSE
NCSAdminVisible=FALSE
DeliveryEngineVisible=FALSE
SubscriptionPortalVisible=FALSE
TutorialDeliveryInstallVisible=FALSE
TutorialDeliveryConfigureVisible=FALSE
SequeLinkVisible=FALSE
PortletsVisible=FALSE
TM1ConnectorVisible=FALSE
GISConnectorsVisible=FALSE

### Components To Install (TRUE) or Remove (FALSE) ###
DesktopAnalystSelect=TRUE
DesktopDesignerSelect=TRUE
ArchitectSelect=FALSE
ProjectBuilderSelect=FALSE
FunctionPluginSelect=FALSE
CommandManagerSelect=FALSE
EnterpriseManagerSelect=FALSE
ObjectManagerSelect=FALSE
SDKSelect=FALSE
ServerAdminSelect=FALSE
IServerSelect=FALSE
IServerOLAPServicesSelect=FALSE
IServerReportServicesSelect=FALSE
IServerDistributionServicesSelect=FALSE
IServerTransactionServicesSelect=FALSE
IntegrityManagerSelect=FALSE
WebAnalystSelect=FALSE
WebProfessionalSelect=FALSE
WebReporterSelect=FALSE
WebServerASPNETSelect=FALSE
WebServerJSPSelect=FALSE
WebServicesASPNETSelect=FALSE
WebServicesJSPSelect=FALSE
OfficeSelect=FALSE
MobileSelect=FALSE
MobileClientSelect=FALSE
MobileServerASPSelect=FALSE
MobileServerJSPSelect=FALSE
TutorialReportingSelect=FALSE
WebMMTSelect=FALSE
AnalyticsModulesSelect=FALSE
NCSAdminSelect=FALSE
DeliveryEngineSelect=FALSE
SubscriptionPortalSelect=FALSE
TutorialDeliveryInstallSelect=FALSE
TutorialDeliveryConfigureSelect=FALSE
SequeLinkSelect=FALSE
PortletsSelect=FALSE
TM1ConnectorSelect=FALSE
GISConnectorsSelect=FALSE

[Programfolder]
HideDialog=TRUE
Name=MicroStrategy

[Summary]
HideDialog=TRUE

[Finish]
HideDialog=TRUE
Copy and paste this example to create a `response.ini` file. Replace any text between angled brackets (<> ) with your own specific information. Make sure to check that all file paths are entered with correct spacing.

Your license key determines which MicroStrategy components will be available for your installation. For example, if your license key does not include MicroStrategy Desktop Designer, then you cannot use `DesktopDesignerVisible=TRUE` and `DesktopDesignerSelect=TRUE` to install these components.

### Using the `response.ini` file

To use the `response.ini` file, do one of the following:

- **Save your `response.ini` file** to the directory `C:\`. You can save to a different directory, but the command below assumes the response file is saved to the directory location `C:\`. Type the following command in the Windows command line:

  ```
  Path\setup.exe --ResponseFile="C:\response.ini"
  ```

- **From the Windows Start menu, choose Run** to open the Run dialog box. Enter the same command in the box and click **OK**.

The setup program supports several command-line parameters. The following applies to this function:

- Parameters using double dashes, such as `--auto`, are defined by MicroStrategy. For example, you can use the `--auto` parameter as follows:

  ```
  Path\setup.exe --Auto=TRUE --LogFile="C:\install.log"
  ```

- The command line is not case-sensitive.

The following parameters are supported by the setup program:

- **Auto**= instructs the setup program to use the response file and default values to enable a one-click installation. If a component, such as serial key or disk space has an invalid value, the setup program automatically reverts to multiple-click mode, and all dialog boxes are displayed.

- **ResponseFile**= contains responses to installation questions and redefines default parameters. The path and file name must be in double quotes (" "). If you use this parameter, do not use any other parameters.

- **ConfigFile**= used by the Configuration Wizard to set up a repository, a server, or a client. The path and file name must be in double quotes (" ").
• **LogFile**= used to specify an alternative location and/or name (other than install.log) for the log file in the Common Files directory. If only the file name is entered, the default location remains the Common Files directory. Once specified, the alternative file becomes the default.

### Activating your installation

After your installation is complete, you must activate your MicroStrategy software installation within 30 days. To activate your software you can follow the instructions provided in *Chapter 4, Activating Your Installation*.

### Configuring your installation with a response.ini file

The Configuration Wizard walks you through the process of setting up the environment for the MicroStrategy products installed in your system. It is possible to configure server definition, project source names, and the metadata repository using a response.ini file. The steps required to create a response.ini file to configure MicroStrategy are provided in *Configuring MicroStrategy with a response file, page 183*.

### Uninstalling with a response.ini file

You can uninstall all MicroStrategy products at once using a response.ini file. You must create a response file with the RemoveAll parameter set to TRUE in the Welcome section. This is also known as a silent uninstallation.

You must save the file as ANSI encoding.

Before uninstallation begins, the MicroStrategy application:

• Checks for user privileges. If they are not valid, uninstallation stops.
• Checks for running components. If one is found, uninstallation stops.
• Stops and deletes the MicroStrategy Intelligence Server service.
• Deletes application created files.

### Example of a response.ini file to uninstall MicroStrategy

You can use the following response file to remove all MicroStrategy products:
[Installer]
PropertiesFilesOverwrite=FALSE
EnableTracing=FALSE
HideAllDialogs=TRUE
ForceReboot=TRUE
PreventReboot=FALSE
CheckTCPIP=TRUE
CheckIIS=TRUE
CheckSP=TRUE
CreateShortcuts=TRUE
CheckRenameOperations=TRUE
AnalyticsOverwrite=FALSE
TutReportingOverwrite=FALSE
TutDeliveryOverwrite=FALSE
BackupFiles=FALSE
RunConfigWizard=FALSE
StopAllServices=TRUE
StopIIS=TRUE
EnableASPServices=FALSE
ConfigWizardResponseFile=
RegistrySizeReq=
LogFile=

[Welcome]
HideDialog=TRUE
RemoveAll=TRUE

[WebVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=MicroStrategy
RemoveVD=YES

[MobileVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=MicroStrategyMobile
RemoveVD=YES

[PortalVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=NarrowcastServer
RemoveVD=YES

[WebServicesDirectory]
HideDialog=TRUE
VirtualDirectory=MicroStrategyWS
RemoveVD=YES
[WebMMTVirtualDirectory]
HideDialog=TRUE
VirtualDirectory=WebMMT
RemoveVD=YES

[Finish]
HideDialog=TRUE

For details on creating a response.ini file, see Creating a response.ini file, page 349.

After you have created a response.ini file, use the following script at the command prompt to uninstall all MicroStrategy products:

```
Path\setup.exe --ResponseFile= “Path\response.ini”
```

where the path for setup.exe must be the path to the original setup.exe used to install MicroStrategy products. The path for the response file is the path where you saved your response.ini file.

**Silent installation**

A silent, or unattended, installation is one that presents no user interface. Silent installations are useful for system administrators who do not want users to interfere with the installation. They are typically implemented by IT departments that perform package-based installations across the network, such as SMS. The silent installation can be used for:

- Deployment with Microsoft System Management Server (SMS)
- OEM installations

Silent installations with MicroStrategy must use a response.ini file in conjunction with a setup.iss file. Using a setup.iss file without a response.ini file is not recommended or supported.
Silent installations through System Management Server environments

Upon setup completion, the Microsoft System Management Server (SMS) generates a Management Information Format (MIF) file. There are two ways to generate the MIF file:

- Using the command line with the switch
  
  \(-m\)FileNameWithoutExtension

- Adding the MIF section to the setup.ini file

Uninstallation in SMS also generates a corresponding MIF file if you use the command line with the switch

\(-m\)FileNameWithoutExtension.

OEM silent installations

You can use silent installation to easily embed MicroStrategy products with other applications. This can be done to develop an OEM application that includes MicroStrategy functionality. For information on performing a silent installation for OEM applications, see OEM silent installations, page 418.

Silent installation output

The system verifies compliance with installation prerequisites and places related messages in a file created for that purpose. The following applies to the generation and storage of output messages during silent installation:

- The MSTRInst.log file is created and placed in the Temp folder.
- The MSTRInst.log file is maintained during the entire setup.
- All system-generated messages, including messages containing reasons for pre-installation termination, are stored in the MSTRInst.log file.

If there are installation termination messages in subsequent installation instances that use the same dialog flow, they are also stored in the MSTRInst.log.
The table below identifies the information that the MSTRInst.log file includes.

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function [F]</td>
<td>Identifies the function calls in the setup script.</td>
</tr>
<tr>
<td>Information [I]</td>
<td>Logs information about the setup that is running.</td>
</tr>
<tr>
<td>Warning [W]</td>
<td>Includes feedback that you must verify related to the setup. For example, in normal mode, when MicroStrategy applications are running on a machine where the setup is being run, you are prompted to close all MicroStrategy applications before proceeding. In silent mode, you are not prompted, and instead the setup terminates.</td>
</tr>
<tr>
<td>Severe [S]</td>
<td>Includes fatal problems that prevent the setup from proceeding. For example, the Intelligence Server Service cannot be created and setup fails as a result.</td>
</tr>
</tbody>
</table>

A typical line in the MSTRInst.log file includes source file name, function name, source line number, and time. It appears as follows:

```
[Z:\InstallMaster\Setup\ScriptFiles\MALicense.rul][UseDLL][l: 28][1318179ms][W] Le file C:\WINDOWS\TEMP\(84D0D5E2-719A-11D7-847C-000C293B5019)\{B339F3B3-E78C-45E8-B4D2-3C46C1C13663}\MAInst.dll couldn't be loaded in memory.
```

When reviewing warning messages in the MSTRInst.log file, look for [W] and [S] to find related problems.

**Activating silent installations**

After your silent installation is complete, you must activate your MicroStrategy software installation within 30 days. To activate your software you can follow the instructions provided in *Chapter 4, Activating Your Installation*.

**Troubleshooting silent installations**

Silent installation may not work if you are installing in a different environment than the one you recorded. This is the case because any dialog box that was not recorded previously is not recognized, such as if you are prompted to stop your Intelligence Server or your IIS Web server. If this happens, check:
• The version of Intelligence Server to ensure that you have the right one for the products you are installing

• You do not have any MicroStrategy applications running

• The setup.log file to see if the ResultCode=0

• Check the install.log for any recorded errors during installation and the MSTRInst.log file for any possible errors

The most common errors are:

• -8, which is an invalid path to the MicroStrategy Installation Wizard silent response .iss file.

• -12, which is dialog boxes out of order. This occurs because an unrecorded screen opened when running the silent install.

It is recommended that you create a silent install that can be used with a response.ini file. For more information, see OEM silent installations, page 399. This way you can change the settings in the response.ini file without having to generate a new .iss file.

Silent installation of a hotfix

A hotfix is a file or a collection of files that you can apply to MicroStrategy products installed on your computer to correct a specific problem. Hotfixes are packaged in an executable (.exe) file, which is a self-installing format. When you install a hotfix, files are backed up automatically so that you can remove the hotfix later.

You can apply a hotfix to MicroStrategy products using silent installation.

Prerequisites

• You must obtain and unzip the MicroStrategy hotfix files. These files include HFResponse.ini and HFsetup.iss. The rest of this procedure assumes you have saved these files to the file path C:\. If you save it to another file path, replace C:\ with the file path to the HFReponse.ini and HFsetup.iss.
To perform a silent installation of a hotfix

If you are installing a hotfix of MicroStrategy Office, see Silent installation of a MicroStrategy Office hotfix, page 402.

1 Insert the disk containing the Hotfix installable in the disk drive.

2 Access the disk drive through the command prompt.

3 Run the silent installation with the following command:

```
InstallPath\setup.exe -LLanguageValue
--ResponseFile="C:\HFresponse.ini" -s
-f1"C:\HFsetup.iss"
```

In the command listed above, you must supply the following information:

- **InstallPath**: The location in which to install the MicroStrategy hotfix.

- **LanguageValue**: Determines the language for the installation. Refer to Language settings for silent installations, page 421, for an explanation of the LanguageValue parameter.

You can check the status of the silent installation in Windows Task Manager. After applying the Hotfix, you can view it in the Add/Remove Programs window.

Silent installation of a MicroStrategy Office hotfix

To perform a silent installation of a MicroStrategy Office hotfix, refer to the procedure below.

To perform a silent installation of a MicroStrategy Office hotfix

1 From a command prompt, navigate to the MicroStrategy Office hotfix installation folder.

2 To view information on the various options of running the silent install command, enter the following command:

```
MicroStrategyOffice.msi /?
```
3 To run the silent installation using installation options defined by the current MicroStrategy Office configuration, enter the following command:

`msiexec.exe /i MicroStrategyOffice.msi /qn`

**Configuring your MicroStrategy installation**

After completing the steps to install Intelligence Server, you can continue the set up and configuration of your installation. To help guide the rest of your installation and configuration steps, refer to the section *Installation and configuration checklists, page 87* in *Chapter 1, Planning Your Installation*, for installation and configuration checklists.
**Introduction**

This chapter explains the various methods of performing a fully automated and unattended installation within the MicroStrategy platform when you do not have access to a UNIX or Linux graphical user interface (GUI).

Intelligence Server Universal configurations possible through the command line on UNIX and Linux are covered in *Chapter 12, Configuring MicroStrategy Using Command Line Tools*.

This chapter includes the following section(s):

- *Silent installation, page 406*

Before installing MicroStrategy products, you should refer to *Chapter 1, Planning Your Installation* for important pre-installation information.
Silent installation

A silent or unattended installation is one that presents no graphical user interface (GUI). Silent installations allow you to automate the installation, so it can be called from a script and executed without user interaction. Silent installations are useful for system administrators who do not want users to run the installation themselves. The silent installation can be done on one or more computers.

Creating an options file

To run silent installation, you first create an options file and then run it with the MicroStrategy Installation Wizard. Save the options file as options.txt.

This options file or response file is used with the command line argument -options to modify the wizard settings. The settings that can be specified for the wizard are listed in the next section, Settings that can be changed, page 407.

To use the options file

1 Log on to the computer where you are installing one or more MicroStrategy products.

2 You can access the installation files and the options file from a disk or you can ask your system administrator to share the files in a network location. For information on mounting or unmounting CD-ROM drives, see Mounting and unmounting CD-ROMs, page 526 in Appendix B, Troubleshooting.

3 Browse to the MicroStrategyInstallation/QueryReportingAnalysis_UNIX directory and open options.txt in a text editor.

4 Specify a value for a setting by replacing the character’s Value. For information on how to specify its value, see Chapter 3, Installing MicroStrategy on UNIX and Linux.

5 Save the changes to the options.txt file.
To use the options file on a silent installation, specify `-silent
-options FileName` as a command line argument to the wizard, where
`FileName` is the name of this options file, for example, `options.txt`.

The following are examples using the native launchers:

- **Solaris**: `setupsol.bin -silent -options options.txt`
- **AIX**: `setupAIX.bin -silent -options options.txt`
- **HP-UX**: `setupHPIA64.bin -silent -options options.txt`

## Settings that can be changed

The following settings can be changed for a silent installation. For details on settings, see *Chapter 3, Installing MicroStrategy on UNIX and Linux*.

The settings follow their descriptions, in the format

`-W settingname=Value`

### Customer information

Your name, the name of the company for which you work, and the license key.

- **User**
  
  `-W userRegistration.user=Value`

- **Company**
  
  `-W userRegistration.company=Value`

- **License key**
  
  `-W userRegistration.cdkey=Value`

### MicroStrategy install locations

The install locations of the product. Specify valid directories where the product should be installed.
- Home directory
  -W silent.homeDirectory=Value

- Install directory
  -W silent.installDirectory=Value

- Log directory
  -W silent.logDirectory=Value

**Product features**

Legal values that you provide for all the product features are:
- True, which indicates that the feature is selected for installation
- False, which indicates that the feature is not selected for installation

**MicroStrategy Intelligence Server**

The selection state of MicroStrategy Intelligence Server.

-P IServerFeature.active=Value

For example, to select MicroStrategy Intelligence Server for installation, use

-P IServerFeature.active=true

**MicroStrategy Web Universal Analyst**

The selection state of MicroStrategy Web Universal Analyst.

-P WebUAnalystFeature.active=Value

For example, to select MicroStrategy Web Universal Analyst for installation, use

-P WebUAnalystFeature.active=true
**MicroStrategy Web Universal Reporter**

The selection state of MicroStrategy Web Universal Reporter.

\[-P\text{ WebUReporterFeature.active}=\text{Value}\]

For example, to select MicroStrategy Web Universal Reporter for installation, use

\[-P\text{ WebUReporterFeature.active}=\text{true}\]

**MicroStrategy Web Universal Professional**

\[-P\text{ WebUProfFeature.active}=\text{Value}\]

For example, to select MicroStrategy Web Universal Professional for installation, use

\[-P\text{ WebUProfFeature.active}=\text{true}\]

**MicroStrategy SDK**

The selection state of MicroStrategy SDK.

\[-P\text{ SDKFeature.active}=\text{Value}\]

For example, to select MicroStrategy SDK for installation, use

\[-P\text{ SDKFeature.active}=\text{true}\]

**CPU license information**

This value should be specified when the license being used for MicroStrategy Intelligence Server is based on CPUs. Legal values are integers between 1 and either the number of CPUs allowed by the license or the number of CPUs in the machine, whichever is lower.

\[-W\text{ cpuCountBean.cpuNumber}=\text{Value}\]

For example, the value

\[-W\text{ cpuCountBean.cpuNumber}=$J(\text{install.cpuCount.maximum})\]

sets this option to the maximum value allowed.
MicroStrategy SDK installation location

The installation location for the MicroStrategy SDK.

-W silent.sdkDirectory="Value"

Override options

The silent installation always overrides files from previous installations.

To preserve some files, do not use a silent installation.

Unique post-installation configurations

MicroStrategy supports a large number of different UNIX and Linux environments with various system configurations. There are a few cases in which you must perform some manual configurations to support the use of MicroStrategy on your system. For more information, refer to Unique post-installation configurations., page 125 in Chapter 3, Installing MicroStrategy on UNIX and Linux.

Silent installation output

The installation returns 0 if the installation is successful and any other value if it is not. The install.log file located in the InstallPath directory provides more information on possible errors. For more information on the install.log file, see Installation log file, page 346 in Chapter 2, Installing MicroStrategy on Windows.

If the installation fails on any of the steps before it starts copying the files, it does not give any feedback other than the return value different from 0.

Activating a silent installation

After your silent installation is complete, you must activate your MicroStrategy installation within 30 days. To activate your installation you can follow the instructions provided in Chapter 4, Activating Your Installation.
Configuring MicroStrategy in command line mode

The MicroStrategy Configuration Wizard is provided in command line mode so that you can use the Configuration Wizard through the operating system console if you do not have access to the GUI. Running the Configuration Wizard in command line mode to configure MicroStrategy on UNIX and Linux machines is covered in the *Configuring MicroStrategy with a response.ini file, page 430* section in *Chapter 12, Configuring MicroStrategy Using Command Line Tools*.

Configuring your MicroStrategy installation

After completing the steps to install MicroStrategy products, you can set up and configure your installation. To help guide the rest of your installation and configuration steps, refer to the section *Installing and configuring MicroStrategy on UNIX and Linux, page 89* in *Chapter 1, Planning Your Installation*, for an installation and configuration checklist.
Introduction

This chapter explains the common workflow for deploying the MicroStrategy platform as an Original Equipment Manufacturer (OEM) application.

The MicroStrategy platform can be deployed as an OEM application in various ways:

• MicroStrategy can be deployed as a software as a service model through the use of MicroStrategy Web. In this scenario MicroStrategy is installed and configured at a centralized location using the standard process, and the customized application is deployed as an OEM application using MicroStrategy Web. For information on deploying MicroStrategy Web, see Chapter 6, Deploying MicroStrategy Web and Web Universal. For information on customizing MicroStrategy Web, see Customizing MicroStrategy Web, page 417.

• MicroStrategy can be deployed as part of an OEM software bundle directly to a customer environment. This chapter focuses on the development and deployment of this type of OEM application.
The table below provides a best practices checklist of how to deploy MicroStrategy as an OEM application.

<table>
<thead>
<tr>
<th>Complete</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Install MicroStrategy on an OEM environment. For information on installing MicroStrategy, see:</td>
</tr>
<tr>
<td></td>
<td>• Chapter 2, Installing MicroStrategy on Windows</td>
</tr>
<tr>
<td></td>
<td>• Chapter 3, Installing MicroStrategy on UNIX and Linux</td>
</tr>
<tr>
<td></td>
<td>Create DSNs using the Connectivity Wizard, as described in Creating DSNs for OEM environments, page 414.</td>
</tr>
<tr>
<td></td>
<td>Configure MicroStrategy using the Configuration Wizard. This tool allows you to save configurations as response files that can be used to automate the configuration for the OEM deployment. This allows you to re-use all the configurations performed when developing an OEM application for the deployment process as well, as described in Configuring a MicroStrategy installation, page 416.</td>
</tr>
<tr>
<td></td>
<td>Design projects and a reporting environment. You can use the various MicroStrategy products and relevant documentation to create the required MicroStrategy environment. For additional best practices when designing a reporting environment, see Designing a project and reporting environment, page 416.</td>
</tr>
<tr>
<td></td>
<td>Customize MicroStrategy Web through the use of the MicroStrategy SDK, as described in Customizing MicroStrategy Web, page 417.</td>
</tr>
<tr>
<td></td>
<td>Deploy a MicroStrategy OEM application on an OEM’s customer environment, as described in Deploying a MicroStrategy OEM application, page 418.</td>
</tr>
<tr>
<td></td>
<td>Create DSNs on the OEM’s customer environment as necessary, as described in Creating DSNs for OEM environments, page 414.</td>
</tr>
<tr>
<td></td>
<td>Configure and tune and OEM deployment through the use of various MicroStrategy tools, as described in Tuning an OEM deployment, page 423.</td>
</tr>
<tr>
<td></td>
<td>If you are modifying a project that has already been deployed as an OEM application, see Updating OEM applications, page 424 for best practices on how to incorporate any custom reports or objects that may have been created for the deployed application.</td>
</tr>
<tr>
<td></td>
<td>Troubleshoot your MicroStrategy OEM applications using MicroStrategy Health Center, as described in Troubleshooting support for MicroStrategy OEM applications, page 426</td>
</tr>
</tbody>
</table>

Creating DSNs for OEM environments

Establishing communication between MicroStrategy and your databases or other data sources is an essential first step in configuring MicroStrategy products for reporting and analysis of your data. These data sources are used to store the data warehouse and the MicroStrategy metadata, which are both required to support a MicroStrategy reporting environment.
To create a connection to these data sources you need an ODBC driver as well as a data source name (DSN). MicroStrategy comes packaged with ODBC drivers to support connecting to various data sources. For more information on ODBC drivers, see *Communicating with databases: ODBC and DSNs, page 144*.

When setting up your OEM environment, you must create a separate DSN to connect to the main data source and the metadata repository. This requirement is true even if the data source and metadata repository are stored in the same database or other data source. The main data source and the metadata are described below:

- A data source stores the data that users of the system can analyze through BI capabilities offered by MicroStrategy products.
- Metadata is a repository whose data associates the tables and columns of a data warehouse with user-defined attributes and facts to enable the mapping of business views, terms, and needs to the underlying database structure. Metadata can reside on the same server as the data warehouse or on a different server. It can be also be stored in a different relational database than your data warehouse. A metadata can be created using the Configuration Wizard, as described in *Configuring a MicroStrategy installation, page 416*.

A DSN can be created using the MicroStrategy Connectivity Wizard, as described in *Defining DSNs, page 151*.

**Creating DSNs as part of an OEM deployment**

As part of the deployment of an OEM application, the Connectivity Wizard can also be run from the command line to create DSNs on UNIX and Linux environments. This allows you to perform this configuration using scripts. For information on creating DSNs using the command line version of the Connectivity Wizard, see *Creating a DSN for a data source, page 428*.

For OEM deployments on Windows machines, use the Connectivity Wizard interface to create DSNs, as described in *Defining DSNs, page 151*.
Configuring a MicroStrategy installation

After installing MicroStrategy, you can use the MicroStrategy Configuration Wizard to configure the metadata repository, statistics tables, History List tables, MicroStrategy Intelligence Server, and multiple project sources.

The Configuration Wizard interface guides you through each of these configurations, as described in Initial MicroStrategy configuration, page 157.

In addition, all configurations that are performed using the Configuration Wizard can be saved as response files. These files can then be used later to automate much of the initial configuration of MicroStrategy when deploying it as an OEM application. This allows you to re-use all the configurations performed when developing an OEM application. For information on configuring MicroStrategy using a response file, see Configuring MicroStrategy with a response file, page 183.

Designing a project and reporting environment

You can use the various MicroStrategy products and relevant documentation to create the required MicroStrategy reporting environment for your OEM application. The following best practices can be helpful when creating this reporting environment:

• It is common to define objects such as reports, documents, attributes, metrics, and filters that are created for the OEM application so that they cannot be modified once it is deployed. You can modify the object security of each object so that it does not allow write access. This ensures that the reports provided out of the box with the OEM application are not modified and overwritten. Users can still use Save As to save their own personal copies of any objects to make any required changes.

• You can modify the folder permissions in MicroStrategy to determine where reports and objects can be created. Limiting the folders that allow write access can require users to create reports in their My Reports folder.

• If you are modifying a project that has already been deployed as an OEM application, see Updating OEM applications, page 424 for best practices on how to incorporate any custom reports or objects that may have been created for the deployed application.
Customizing MicroStrategy Web

MicroStrategy Web provides users with a highly interactive environment and a low-maintenance interface for reporting and analysis. Using the MicroStrategy Web interface, users can access, analyze, and share corporate data through any web browser on any operating system.

With the MicroStrategy SDK, you can customize, embed, or extend MicroStrategy Web into your application, or modify the standard interface or functionality. Common customizations for OEM deployments include:

- Customizing the look and feel of the MicroStrategy Web interface. This can include changing the color scheme, adding or removing content, using customized logos, and many other customizations.
- Integrating MicroStrategy Web with third-party applications such as:
  - Portals
  - External security and user management systems
  - Advanced data visualizations
- Extending functionality to support composite applications, include writeback capabilities, and implementing other custom features.

To customize MicroStrategy Web using the MicroStrategy SDK, refer to the MicroStrategy Developer Library (MSDL). The MSDL contains details about the architecture, object models, customization scenarios, code samples, and so on that are useful for building a sophisticated and highly functional, customized application.

To access the MSDL, you must own a MicroStrategy SDK license. The MSDL is installed in the docs subfolder within your MicroStrategy SDK installation:

- In Windows: From the Windows Start menu, point to Programs, then MicroStrategy, then SDK, and then select Developer Library.
- In UNIX: Navigate to the SDK/docs directory and double-click DeveloperLibrary.htm to launch the Welcome page of the MSDL.
Deploying a MicroStrategy OEM application

Once an OEM application is developed, it then must be deployed to the customer’s environment. Steps to deploy an OEM installation include:

1. Installing the required MicroStrategy products on the customer’s environment. This can be automated using silent installation techniques, as described in OEM silent installations, page 418. To use the Installation Wizard to install MicroStrategy products, see:
   - Chapter 2, Installing MicroStrategy on Windows
   - Chapter 3, Installing MicroStrategy on UNIX and Linux

2. Additional configurations are required, as described in Configuring an OEM deployment installation, page 422.

OEM silent installations

You can use silent installation to easily embed MicroStrategy products with other applications. The steps below show you how to use a silent installation to deploy an OEM application on a Windows environment. For additional information on silent installations, see Chapter 9, Automated Installation on Windows.

To perform a silent installation on a UNIX or Linux environment, see Silent installation, page 406.

To perform an OEM silent installation

When MicroStrategy products are installed as software bundled with another product, the following procedure is strongly recommended:

1. Create an installation response file (response.ini) for the MicroStrategy products to install. The table that follows shows which sections of the file are mandatory and which are optional.
For detailed information regarding the contents of the response.ini file, see "Configuring a response.ini file to install MicroStrategy, page 348."

<table>
<thead>
<tr>
<th>Response File Section</th>
<th>Selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Installer]</td>
<td>Required</td>
</tr>
<tr>
<td>HideAllDialogs =</td>
<td>Required</td>
</tr>
<tr>
<td>PreventReboot =</td>
<td>Optional</td>
</tr>
<tr>
<td>StopAllServices =</td>
<td>Optional</td>
</tr>
<tr>
<td>StopIIS =</td>
<td>Optional</td>
</tr>
<tr>
<td>CheckRenameOperations =</td>
<td>Optional</td>
</tr>
<tr>
<td>[UserRegistration]</td>
<td>Required</td>
</tr>
<tr>
<td>[ComponentSelection]</td>
<td>Required</td>
</tr>
<tr>
<td>EnterpriseManagerSelect =</td>
<td>Required</td>
</tr>
<tr>
<td>[InitialPaths]</td>
<td>Required</td>
</tr>
<tr>
<td>EnterpriseManager =</td>
<td>Required</td>
</tr>
</tbody>
</table>

Setting HideAllDialogs = TRUE causes the script for the response file to:

- Use default values as specified in the response.ini file.
- Not require user input.
- Keep the dialog flow consistent from one instance to the next. Consistency in the response file script from one instance to the next is necessary; if setup.iss detects an inconsistency in the dialog flow, installation is terminated and a log file for the failure is created.

The only dialog flow modifications pertinent to silent installation are specific to file location. Therefore, the only portion of the response.ini that may need to be modified is the [InitialPaths] section.

The rest of this procedure assumes you have saved the response.ini file to the file path C:\. If you save it to another file path, replace C:\response.ini with the file path of your response.ini file.

You must save the response.ini file as ANSI encoding.
2 Create the `setup.iss` file to use in conjunction with the `response.ini` file for the silent installation. Use a text editor to create the `setup.iss` file with the following information:

```
[InstallShield Silent]
Version=v7.00
File=ResponseFile

[File Transfer]
OverwrittenReadOnly=NoToAll

[Application]
Name=MicroStrategy
Version=x.y.z
#x.y.z represent the version of the product#
Company=MicroStrategy
Lang=LanguageValue
[{8CCF3F6C-55B7-4A27-8C68-ADF21D0585A2}]
-DlgOrder
Count=0
```

You must save the `setup.iss` file as ANSI encoding.

Be aware of the following

- The version in the `setup.iss` file must match the MicroStrategy version you are installing exactly. For example, if you are installing version 9.2.1 you must enter `Version=9.2.1`. Entering a version as `Version=9` or `Version=9.2.x` causes an error when trying to perform a silent installation of version 9.2.1.

- Refer to Language settings for silent installations, page 421, for an explanation of the `LanguageValue` parameter within the line `Lang=LanguageValue`.

3 Run the silent install with the `response.ini` file in conjunction with the `setup.iss` file as follows:

Refer to Language settings for silent installations, page 421, for an explanation of the `LanguageValue` parameter.

```
INSTALL_PATH\setup.exe -LLanguageValue
--ResponseFile="C:\response.ini" -s -f1"c:\setup.iss"
-f2"c:\setup.log"
```

In the syntax shown above, the `-s` parameter indicates that the installation is to be completely silent. If the `-s` parameter is not included
in the command, then an interface is displayed during the installation that shows the progress of the installation.

If the setup program encounters an invalid value for an installation requirement, the setup terminates and the silent installation is ended. You can review any errors in the setup.log file.

4 If a restart is required after the installation is complete, a restart of the machine is automatically triggered. Power the machine back on to allow for the completion of any configurations that are required after the restart of the machine.

5 After the installation is complete, you can check the result of the installation process. If the silent installation is successful, the resulting code value is zero (ResultCode=0) in the setup.log file. This is the only indication of the installation being completed if the installation is completely silent and a restart of the machine is not required.

Language settings for silent installations

In the final two steps of the procedure to run an OEM silent installation:

• You can set Desktop language settings by setting the language value in the setup.iss file.
• You can bypass the language prompt by running setup.exe with the command line option for the language.

The following table lists the values for the different languages that MicroStrategy supports.

<table>
<thead>
<tr>
<th>Language</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Danish</td>
<td>0006</td>
</tr>
<tr>
<td>Dutch</td>
<td>0019</td>
</tr>
<tr>
<td>English</td>
<td>0009</td>
</tr>
<tr>
<td>French</td>
<td>0012</td>
</tr>
<tr>
<td>German</td>
<td>0007</td>
</tr>
<tr>
<td>Italian</td>
<td>0016</td>
</tr>
<tr>
<td>Japanese</td>
<td>0017</td>
</tr>
<tr>
<td>Korean</td>
<td>0018</td>
</tr>
</tbody>
</table>
For example, to select English as the language:

- **For the setup.iss file**, change `Lang=LanguageValue` to:

  ```
  Lang=0009
  ```

- **To run the silent install**, use the command line option as follows:

  ```
  Path\setup.exe -L0009
  ```

  For the command line option, you must type `-L` in front of the language code to signify that you are entering a language.

### Configuring an OEM deployment installation

Once MicroStrategy has been installed on the customer’s environment, the following additional steps must be taken to prepare the initial configuration of MicroStrategy software:

- Provide the MicroStrategy metadata that was developed for the OEM application.

- Create separate DSNs to connect to the data warehouse and the metadata, as described in *Creating DSNs for OEM environments*, page 414.

- Use the Configuration Wizard to configure metadata, Intelligence Server, and project sources. If you saved your configurations as response files, these configurations can be re-used for automated configuration. For information on using the Configuration Wizard, see *Configuring a MicroStrategy installation*, page 416.

- Configure and tune an OEM deployment through the use of various MicroStrategy tools, as described in *Tuning an OEM deployment*, page 423.
Tuning an OEM deployment

A MicroStrategy OEM deployment requires additional tuning and configuration, both during deployment and throughout the life cycle of the OEM application. Various ways to perform these configurations are described below:

• Tuning with Command Manager, page 423
• Configuring MicroStrategy in command line mode, page 424

Tuning with Command Manager

MicroStrategy Command Manager lets you perform various administrative and application development tasks by using text commands that can be saved as scripts. You can manage configuration settings within the MicroStrategy platform, for either project sources or Narrowcast Server metadatas. With Command Manager, you can change multiple configuration settings all at once, without using the MicroStrategy Desktop or Narrowcast Administrator interface.

Developers of OEM applications that use embedded MicroStrategy projects may find that they need flexibility in configuring their environment. Command Manager Runtime is a slimmed-down version of the Command Manager command-line executable for use with these OEM applications. For information about obtaining Command Manager Runtime, contact your MicroStrategy sales representative.

OEM application deployments typically required on-premise configuration of environment-specific settings such as database user and password, governing options, caching options, and other tuning requirements. Command Manager Runtime scripts enable OEMs to automate a number of such configuration settings.

Command Manager Runtime uses a subset of the commands available for 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.” For a list of the commands available in Command Manager Runtime, with syntax and examples for each command, refer to the Supplemental Reference for System Administration.
Configuring MicroStrategy in command line mode

The MicroStrategy Configuration Wizard is provided in command line mode so that you can use the Configuration Wizard through the operating system console if you do not have access to the GUI. Running the Configuration Wizard in command line mode to configure MicroStrategy on UNIX and Linux machines is covered in Configuring MicroStrategy with a response.ini file, page 430.

Updating OEM applications

The life-cycle of an OEM application often requires the OEM application to be updated with new reports and other enhancements. These enhancements can be developed within the OEM application and then the customer’s existing application can be updated.

For more information on updating OEM applications, see:

- Modifying deployed OEM applications, page 424
- Deploying an OEM application update, page 425

Modifying deployed OEM applications

If you are modifying a project that has already been deployed as an OEM application, you must update the application in a way that does not disrupt any current customer development, as described below:

- Retrieve the customer’s metadata so that custom reports or other objects that have been created can also be included in the OEM application update.

- Any new objects deployed as part of the OEM application update should be tested to ensure that they do not negatively affect the objects provided with the previous deployment of the OEM application. MicroStrategy Integrity Manager can be used to automate the testing of including new objects in an OEM application. For information on Integrity Manager, refer to the System Administration Guide.
Deploying an OEM application update

Once updates for an OEM application are complete, the OEM application must then be deployed. There are two ways in which an updated OEM application can be redeployed, as described below:

- Replace the entire project in the OEM application, as described in *Replacing a project in an OEM application, page 425*.

- Merge new objects into the existing project for the OEM application, as described in *Merging new objects into a project in an OEM application, page 425*.

Replacing a project in an OEM application

Once updates for an OEM application are complete, the project can be replaced in an OEM application. This is commonly done by duplicating the updated OEM project, and then merging that project into the production OEM application.

These tasks can be achieved using the MicroStrategy Project Duplication Wizard and Project Merge Wizard. Both of these tools can perform their tasks from the command line, which can allow the project duplication and replacement process to be automated. The steps to use these tools to duplicate and replace a project are provided in the *System Administration Guide*.

Merging new objects into a project in an OEM application

Once updates for an OEM application are complete, the updates to the project can be deployed through the use of update packages. An update package is a file containing a set of object definitions and conflict resolution rules. When you create an update package, you first add objects, and then specify how any conflicts involving the objects are resolved.

These update packages can be developed using Object Manager. Once the package is ready for deployment, it can be deployed using Object Manager or Command Manager Runtime. For information on creating and deploying an update package with Object Manager, refer to the information on managing projects provided in the *System Administration Guide*. For information on Command Manager Runtime, refer to the *Supplemental Reference for System Administration*. 
Troubleshooting support for MicroStrategy OEM applications

MicroStrategy Health Center can help you diagnose and fix problems in your MicroStrategy system. It detects known problems and provides an immediate solution. In cases where Health Center cannot fix a problem immediately, it enables you to bundle relevant log files into a diagnostic package and transmit this package to MicroStrategy Technical Support for review and troubleshooting.

Health Center is provided with a MicroStrategy installation.

For information on using Health Center to diagnose and fix problems in your MicroStrategy environment, refer to the System Administration Guide.
Introduction

MicroStrategy tools are provided in command line mode on UNIX and Linux so that you can perform various configuration tasks through the operating system console. This enables you to perform your required configurations even if you do not have access to the MicroStrategy interface.

On a Windows machine, it is recommended to use the appropriate MicroStrategy interfaces to perform the configurations described in this chapter.

When you perform MicroStrategy configuration tasks through the UNIX or Linux operating system console, you must make sure reserved words and characters are not mistakenly included in your commands. To avoid issues caused by reserved words and characters, see Supporting reserved words and characters, page 453.

This chapter covers the configurations listed below:

- Creating a DSN for a data source, page 428
- Testing and examining ODBC connectivity, page 429
Creating a DSN for a data source

After you install an ODBC driver (see Appendix A, Connecting to Databases: ODBC and DSNs), you can define one or more data sources for it. The DSN should provide a unique description of the data, for example, Payroll_Project_Metadata or Payroll_Warehouse.

The DSN is the name for a pointer used by a client application (in this case MicroStrategy) to find and connect to a data source. Multiple DSNs can point to the same data source and one DSN can be used by different applications.

MicroStrategy provides a one-line command line version of the MicroStrategy Connectivity Wizard to create DSNs on UNIX and Linux.

You can create DSNs using the MicroStrategy Connectivity Wizard on Windows, UNIX, and Linux machines, as described in Communicating with databases: ODBC and DSNs, page 144.

To create a DSN on UNIX/Linux from the command line

1. From a UNIX/Linux console window, browse to HOME_PATH, where HOME_PATH is the directory that you specified as the home directory during installation.

2. Browse to the folder bin.

3. Type mstrconnectwiz -h, and then press ENTER to display command line syntax and examples for different database platforms.

4. Create your command based on the syntax and examples displayed. For example, the command below creates a DSN for an Oracle database and tests login credentials:

   mstrconnectwiz ORCLW MyOracleDSN 12.34.56.78 orcl 1521 -u:OracleUser -p:OracleUserPassword
Testing and examining ODBC connectivity

For UNIX and Linux, the DB Query Tool (mstrdbquerytool) was created by MicroStrategy and is installed with MicroStrategy for UNIX and Linux in the target directory. A command line version of the DB Query Tool (mstrtodbcx) is also available in the same directory.

This tool can test ODBC connectivity to a DSN, list tables available for a DSN, list all DSNs for an ODBC connection, and return other helpful information.

You can also use the DB Query Tool interface on Windows, UNIX, and Linux machines, as described in Using the DB Query Tool, page 155.

To test and examine ODBC connectivity

1. In a UNIX/Linux console window, browse to HOME_PATH, where HOME_PATH is the directory that you specified as the home directory during installation.

2. Browse to the folder bin.

3. Type mstrtodbcx, and then press ENTER. A welcome message is displayed.

4. Type .help, and then press ENTER to display command line syntax and available options.

5. To connect to a DSN, enter the command below:
   
   .cn -d DSN -u UserName -p Password

   In the syntax above, DSN is the data source name, UserName is the login name for the data source, and Password is the password for the data source login. To continue the example from To create a DSN on UNIX/Linux from the command line, page 428, connect to the Oracle DSN with the following command:

   .cn -d MyOracleDSN -u OracleUser -p OracleUserPassword

   Basic connection information is displayed for the successful connection.

6. Run any required tests and options. For example, type .lt -t USER% and press ENTER to list all tables that start with USER. The % symbol is used as a wildcard character in this command.
7 Once you are finished testing and review ODBC connectivity, type `.disconnect` and press ENTER to close the connection.

8 Type `.quit` and press ENTER to close the MicroStrategy DB Query Tool.

## Configuring MicroStrategy with a response.ini file

The MicroStrategy Configuration Wizard is provided in command line mode so that you can use the Configuration Wizard even if you do not have access to a GUI. You can perform the following configurations with the Configuration Wizard in command line mode:

- Create metadata and statistics tables
- Create new MicroStrategy Intelligence Server definitions in the metadata, assign an existing server definition for Intelligence Server, and delete existing server definitions
- Create MicroStrategy project sources in a server (three-tier) mode

Direct (two-tier) data sources are available only on the Windows operating system.

Using the Configuration Wizard in command line mode creates a `response.ini` file. This file can then be used from the command line to configure MicroStrategy without stepping through the pages of the Configuration Wizard. You can also distribute a `response.ini` file to other users and machines to perform multiple configurations without stepping through the Configuration Wizard for each configuration.

This section covers the following procedures and information related to configuring MicroStrategy from the command line on a Linux or UNIX machine using a `response.ini` file:

- *Creating a response.ini file, page 431*
- *Using the response.ini file to configure MicroStrategy, page 448*
- *Parameters and options in the response.ini file, page 448*

You can also configure MicroStrategy using the Configuration Wizard in command line mode on a Windows machine. However, on a Windows machine, it is recommended to use the Configuration Wizard.
Wizard graphical user interface to create and use a response file, which is described in *Configuring MicroStrategy with a response file*, page 183.

Before you can configure MicroStrategy with the Configuration Wizard in command line mode, you must ensure that you meet the prerequisites listed in *Configuration Wizard prerequisites*, page 158.

**Creating a response.ini file**

This section describes how to configure MicroStrategy using the command line mode. Performing the steps in this section creates a *response.ini* file that can be used to configure MicroStrategy installations on UNIX and Linux machines.

---

**To configure MicroStrategy using the command line mode**

1. In a UNIX or Linux console window, browse to *HOME_PATH* where *HOME_PATH* is the directory you specified as the Home Directory during installation.

2. Browse to the *bin* directory.

3. At the command prompt, type `mstrcfgwiz-editor`, then press ENTER. The Configuration Wizard opens in command line mode.
   
   The sections or pages of the wizard displayed depend on your selections.

4. Review the information on the welcome screen and press ENTER to continue.

5. You can select to use a *response.ini* file to configure MicroStrategy, or create a new *response.ini* file to support a new configuration, as described below:

   - Type 1, and then press ENTER to use a *response.ini* file to configure MicroStrategy. For steps to use a *response.ini* file in command line mode, see *Using the response.ini file to configure MicroStrategy*, page 448.

   - Type 2, and then press ENTER to create a new *response.ini* file. You can select from various configuration tasks, as described in the *Configuration tasks* section within this procedure.
Configuration tasks

You can support the configuration tasks described in the sections listed below:

- Type 1, and then press ENTER to create metadata, History List, and statistics tables. Refer to Creating metadata, History List, and statistics tables, page 432 for steps to create metadata and statistics tables.
- Type 2, and then press ENTER to configure a MicroStrategy Intelligence Server definition. Refer to Setting up MicroStrategy Intelligence Server, page 441 for steps to configure an Intelligence Server definition.
- Type 3, and then press ENTER to create project sources. Refer to Creating a project source, page 446 for steps to create project sources.

Creating metadata, History List, and statistics tables

If you selected option 1 in Configuration tasks, page 432, you can create metadata tables, History List tables, and statistics tables. The steps to perform these configuration tasks are provided separately in the sections below:

- Creating metadata tables, page 432
- Creating History List tables
- Creating statistics tables, page 439

Creating metadata tables

You can create metadata tables in a data source, as described in the procedure below.

If metadata tables already exist in the location you plan to store your metadata tables in and you do not want to overwrite the current metadata tables, you should use the option described below.

Prerequisite

- This procedure assumes you have already opened the Configuration Wizard in command line mode and selected to create metadata and statistics tables, as described in Creating a response.ini file, page 431.
To create metadata tables

1 In the prompt asking whether to create metadata tables, type \textit{Y}, and then press \texttt{ENTER}. You are then prompted for ODBC data source information.

2 Type the number corresponding to the ODBC DSN for the database to store your metadata tables, and then press \texttt{ENTER}.

   \textbf{If you do not have a DSN defined on your UNIX or Linux machine, see \textit{Creating a DSN for a data source}, page 428.}

3 If the Configuration Wizard detects an existing metadata repository in the database location you specified, a message is displayed on whether to re-create the metadata tables. If you type \textit{Y} and press enter, all information in the existing metadata repository is overwritten when the response file is executed at a later time.

4 Depending on your database type, you may be prompted to provide a login and password to your DSN:

   a Type a login name for your database that stores your metadata tables, and then press \texttt{ENTER}. You are then prompted to provide a password for the login name.

   b Type a password for the login name provided, and then press \texttt{ENTER}. You are then prompted to provide a metadata prefix for the metadata tables.

5 Depending on your database type, you can enter characters to use as a prefix for the names of your metadata tables or use no prefix, as described below:

   \begin{itemize}
   \item Type the required prefix characters, and then press \texttt{ENTER}.
   \item Leave the prompt blank, and then press \texttt{ENTER} to provide no metadata prefix.
   \end{itemize}

6 The next configuration displayed depends on your ODBC data source details:

   \begin{itemize}
   \item If the data source points to a DB2 MVS database, steps to configure a DB2 MVS database are displayed. These are described in the \textit{To configure DB2 MVS database options} section within this procedure.
   \end{itemize}
• If the data source does not point to a DB2 MVS database, the step to select a metadata script is displayed. This step is described in the To select a metadata script, page 434 section within this procedure.

To configure DB2 MVS database options

These steps are displayed if you are creating your metadata tables in a DB2 MVS database.

7 You can enter the database name to use or use the default name, as described below:
   • Type the database name to use, and then press ENTER.
   • Leave the prompt blank, and then press ENTER to use the default database.

You are then prompted to provide the MVS table space name.

8 You can enter characters to use as a table space name for your metadata tables or use the default table space name, as described below:
   • Type the required table space name characters, and then press ENTER.
   • Leave the prompt blank, and then press ENTER to use the default table space name.

You are then prompted to select a metadata script to create the metadata tables.

To select a metadata script

9 You can select the script used to create the metadata tables or use the default script, as described below:
   • Enter a valid path to a script file, and then press ENTER.
   • Leave the field blank, and then press ENTER to use the default script for your database type.

You are then prompted to create History List tables.

To create History List tables

10 You can choose whether to create History List tables or not, as described below:
   • Type Y, and then press ENTER to create History List tables. Creating History List tables is described in Creating statistics tables, page 439.
• Type N, and then press ENTER to skip History List table creation. You are then prompted to create statistics tables, as described in the To create statistics tables section within this procedure.

To create statistics tables

11 You can choose whether to create statistics tables or not, as described below.

• Type Y, and then press ENTER to create statistics tables. Creating statistics tables is described in Creating statistics tables, page 439.

• Type N, and then press ENTER to skip statistics tables creation. You are then prompted to provide a name for the response.ini file, as described in the To generate a response.ini file section within this procedure.

To generate a response.ini file

12 By default, the configuration is saved as Response.ini in the /HOME_PATH/ directory, where HOME_PATH is the directory you specified as the Home Directory during installation. You can leave the field blank to use the default name or type a different name, and then press ENTER. The response.ini file is generated, and you are prompted whether to run the configuration immediately.

13 You can choose to run the configuration you just completed or to run the configuration using the response.ini file at a different time, as described below:

• Type Y, and then press ENTER to run the configuration.

You can also use the response.ini file created for future configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

• Type N, and then press ENTER to quit without running the configuration. You can use the response.ini file created for future configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

Creating History List tables

You can create History List tables in a data source, as described in the procedure below.
Prerequisite

- This procedure assumes you have already opened the Configuration Wizard in command line mode and selected to create metadata, History List, and statistics tables, as described in Creating a response.ini file, page 431.

To create History List tables

1. After you create metadata tables (see Creating metadata tables, page 432) or skip both of the creation of metadata tables, you are prompted to create History List tables.

   To create History List tables, type Y, and then press ENTER. You are then prompted for ODBC data source information.

2. You can supply ODBC DSN information in various ways described below, which depend on whether you previously created metadata tables as part of the configuration process:

   - If you did not create metadata tables as part of the configuration process, you are prompted to enter ODBC DSN information. The steps to enter this information is described in the To provide ODBC DSN information section within this procedure.

   - If you created metadata tables as part of the configuration process, you are prompted whether to use the same metadata table ODBC DSN information for your History List tables. You have the following options:

     - Type Y, and then press ENTER to create History List tables with the same ODBC DSN information entered for your metadata tables. You are then prompted to select a History List script to create the History List tables, which is described in the To select a History List script section within this procedure.

     - Type N, and then press ENTER to provide different ODBC DSN information, which is described in the To provide ODBC DSN information section within this procedure.
To provide ODBC DSN information

3 Type the number corresponding to the ODBC DSN for a database to create your History List tables in, and then press ENTER.

If you do not have a DSN defined on your UNIX or Linux machine, see Creating a DSN for a data source, page 428.

4 Depending on your database type, you may be prompted to provide a login and password to your DSN:

   a Type a login name for your database to create your History List tables in, and then press ENTER. You are then prompted to provide a password for the login name.

   b Type a password for the login name provided, and then press ENTER.

5 The next configuration displayed depends on your ODBC data source details:

   • If the data source points to a DB2 MVS database, steps to configure a DB2 MVS database are displayed. These are described in the To configure DB2 MVS database options section within this procedure.

   • If the data source does not point to a DB2 MVS database, the step to select a statistics script to create statistics tables is displayed. This step is described in the To select a History List script section within this procedure.

To configure DB2 MVS database options

These steps are displayed if you are creating your metadata tables in a DB2 MVS database.

6 You can enter the database name to use or use the default name, as described below:

   • Type the database name to use, and then press ENTER.

   • Leave the prompt blank, and then press ENTER to use the default database.

You are then prompted to provide the MVS table space name.

7 You can enter characters to use as a table space name for your metadata tables or use the default table space name, as described below:

   • Type the required table space name characters, and then press ENTER.
• Leave the prompt blank, and then press ENTER to use the default table space name.

You are then prompted to select a statistics script to create statistics tables.

To select a History List script

8 You can select the script used to create the History List tables or use the default script, as described below:

• Enter a valid path to a script file, and then press ENTER.

• Leave the field blank, and then press ENTER to use the default script for your database type.

You are then prompted to create statistics tables.

To create statistics tables

9 You can choose whether to create statistics tables or not, as described below.

• Type Y, and then press ENTER to create statistics tables. Creating statistics tables is described in Creating statistics tables, page 439.

• Type N, and then press ENTER to skip statistics tables creation. You are then prompted to provide a name for the response.ini file, as described in the To generate a response.ini file section within this procedure.

To generate a response.ini file

10 By default, the configuration is saved as Response.ini in the /HOME_PATH/ directory, where HOME_PATH is the directory you specified as the Home Directory during installation. You can leave the field blank to use the default name or type a different name, and then press ENTER. The response.ini file is generated, and you are prompted whether to run the configuration immediately.

11 You can choose to run the configuration you just completed or to run the configuration using the response.ini file at a different time, as described below:

• Type Y, and then press ENTER to run the configuration.

You can also use the response.ini file created for future configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.
• Type N, and then press ENTER to quit without running the configuration. You can use the response.ini file created for future configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

Creating statistics tables

You can create statistics tables in a data source, as described in the procedure below.

Prerequisites

• This procedure assumes you have already opened the Configuration Wizard in command line mode and selected to create metadata, History List, and statistics tables, as described in Creating a response.ini file, page 431.

To create statistics tables

1 After you create metadata tables (see Creating metadata tables, page 432), create History List tables (Creating History List tables, page 435), or skip both of these procedures, you are prompted to create statistics tables.

To create statistics tables, type Y, and then press ENTER. You are then prompted for ODBC data source information.

2 You can supply ODBC DSN information in various ways described below, which depend on whether you previously created metadata tables as part of the configuration process:

• If you did not create metadata tables as part of the configuration process, you are prompted to enter ODBC DSN information. The steps to enter this information is described in the To provide ODBC DSN information section within this procedure.

• If you created metadata tables as part of the configuration process, you are prompted whether to use the same metadata table ODBC DSN information for your statistics tables. You have the following options:

   – Type Y, and then press ENTER to create statistics tables with the same ODBC DSN information entered for your metadata tables. You are then prompted to select a statistics script to create the statistics tables, which is described in the To select a statistics script section within this procedure.
– Type N, and then press ENTER to provide different ODBC DSN information, which is described in the To provide ODBC DSN information section within this procedure.

To provide ODBC DSN information

3 Type a valid ODBC DSN for a database to create your statistics tables in, and then press ENTER. You are then prompted to provide a login to your DSN.

4 Type a login name for your database to create your statistics tables in, and then press ENTER. You are then prompted to provide a password for the login name.

5 Type a password for the login name provided, and then press ENTER.

The next configuration displayed depends on your ODBC data source details:

• If the data source points to a DB2 MVS database, steps to configure a DB2 MVS database are displayed. These are described in the To configure DB2 MVS database options section within this procedure.

• If the data source does not point to a DB2 MVS database, the step to select a statistics script to create statistics tables is displayed. This step is described in the To select a statistics script section within this procedure.

To configure DB2 MVS database options

These steps are displayed if you are creating your metadata tables in a DB2 MVS database.

6 You can enter the database name to use or use the default name, as described below:

• Type the database name to use, and then press ENTER.

• Leave the prompt blank, and then press ENTER to use the default database.

You are then prompted to provide the MVS table space name.

7 You can enter characters to use as a table space name for your metadata tables or use the default table space name, as described below:

• Type the required table space name characters, and then press ENTER.
• Leave the prompt blank, and then press ENTER to use the default table space name.

You are then prompted to select a statistics script to create statistics tables.

**To select a statistics script**

8 You can select the script used to create the statistics tables or use the default script, as described below:

• Enter a valid path to a script file, and then press ENTER.

• Leave the field blank, and then press ENTER to use the default script for your database type.

You are then prompted to provide a name for the response.ini file.

**To generate a response.ini file**

9 By default, the configuration is saved as Response.ini in the /HOME_PATH/ directory, where HOME_PATH is the directory you specified as the Home Directory during installation. You can leave the field blank to use the default name or type a different name, and then press ENTER. The response.ini file is generated, and you are prompted whether to run the configuration immediately.

10 You can choose to run the configuration you just completed or to run the configuration using the response.ini file at a different time, as described below:

• Type Y, and then press ENTER to run the configuration.

  You can also use the response.ini file created for future configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

• Type N, and then press ENTER to quit without running the configuration. You can use the response.ini file created for future configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

**Setting up MicroStrategy Intelligence Server**

If you selected option 2 in Configuration tasks, page 432, you can set up MicroStrategy Intelligence Server to create, use, or delete server definitions.
To begin setting up your server definition, you must enter information about your ODBC DSN and MicroStrategy connections.

To set up MicroStrategy Intelligence Server

1. Type the corresponding number for the ODBC DSN for a database to connect Intelligence Server to. This should be the data source that stores your metadata. Then press **ENTER**. You are then prompted to provide a login to your DSN.

2. Type a login name for your database to create your statistics tables in, and then press **ENTER**. You are then prompted to provide a password for the login name.

3. Type a password for the login name provided, and then press **ENTER**. You are then prompted to provide a metadata prefix.

4. You can enter characters to use as a prefix for the names of your metadata tables or use no prefix, as described below:
   - Type the required prefix characters, and then press **ENTER**.
   - Leave the prompt blank, and then press **ENTER** to provide no metadata prefix.

   You are then prompted to provide a temp table prefix.

5. You can enter characters to use as a prefix for the names of temp tables or use no prefix, as described below:
   - Type the required prefix characters, and then press **ENTER**.
   - Leave the prompt blank, and then press **ENTER** to provide no temp table prefix.

   You are then prompted to provide a MicroStrategy user login.

6. Type a valid MicroStrategy user login that has administrator privileges, and then press **ENTER**. You are then prompted to provide a password for the login name.

   The default administrator account is **Administrator** with a blank password. This should be changed after you initial configuration.
7 Type a password for the MicroStrategy user login provided, and then press ENTER. You are then prompted to choose the type of Intelligence Server configuration to complete.

8 You can perform one of the Intelligence Server configuration tasks, which are described in the sections below:

- Type 1, and then press ENTER to create a new server definition. Refer to Creating and using a server definition, page 443 for steps to create a new server definition.

- Type 2, and then press ENTER to use an exiting server definition. This configuration follows the same steps for creating a new server definition, which are described in Creating and using a server definition, page 443.

- Type 3, and then press ENTER to delete a server definition. Refer to Deleting a server definition, page 445 for steps to delete a server definition.

- Type 4, and then press ENTER to create a new server definition and use it as the default server definition. This configuration follows the same steps for creating a new server definition, which are described in Creating and using a server definition, page 443.

Creating and using a server definition

You perform the same steps to complete the following actions:

- Create a new server definition
- Create a new server definition and use it as the default server definition
- Use an existing server definition.

The action taken depends on what action you selected to complete in the procedure To set up MicroStrategy Intelligence Server, page 442.

Prerequisites

- This procedure assumes you have already opened the Configuration Wizard in command line mode and entered configuration information for your Intelligence Server, as described in Setting up MicroStrategy Intelligence Server, page 441.
To create and use a server definition

1 In the prompt that asks for a server definition name, type the name that distinguishes the server definition, and press ENTER. You can press ENTER without entering any information to use the default server definition. You are then prompted to choose the projects to load for the server definition.

2 Type the names of projects to load when the server definition starts, and then press ENTER. Separate the project names with the \ character. You are then prompted to choose projects to not load for the server definition.

3 Type the names of projects to not load when the server definition starts, and then press ENTER. Separate the project names with the \ character. You are then prompted to provide a TCP port to use for Intelligence Server.

4 You can use the default port number or enter a different port number for Intelligence Server, as described below:
   • Leave the prompt blank, and then press ENTER to use the default port number.
   • Type a port number, and then press ENTER.
You are then prompted whether to register Intelligence Server as a service.

5 You can choose whether to register Intelligence Server as a service, as described below:
   • Type Y, and then press ENTER to register Intelligence Server as a service. To perform this configuration, you must be logged into your UNIX or Linux machine with an account that has root level access and permissions.
   • Type N, and then press ENTER to not register Intelligence Server as a service.
You are then prompted whether to start Intelligence Server when finished

6 Type y and press ENTER to start Intelligence Server after the response file is executed. You are then prompted to provide a name for the response.ini file.
To generate a response.ini file

7 By default, the configuration is saved as Response.ini in the /HOME_PATH/ directory, where HOME_PATH is the directory you specified as the Home Directory during installation. You can leave the field blank to use the default name or type a different name, and then press ENTER. The response.ini file is generated, and you are prompted whether to run the configuration immediately.

8 You can choose to run the configuration you just completed or to run the configuration using the response.ini file at a different time, as described below:

   • Type Y, and then press ENTER to run the configuration.

     You can also use the response.ini file created for future configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

   • Type N, and then press ENTER to quit without running the configuration. You can use the response.ini file created for future configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

Deleting a server definition

You can delete a server definition to remove it from the available server definitions for Intelligence Server.

Prerequisites

   • This procedure assumes you have already opened the Configuration Wizard in command line mode and entered configuration information for your Intelligence Server, as described in Setting up MicroStrategy Intelligence Server, page 441.

To delete a server definition

1 In the prompt that asks for server definitions to be removed, type the name that distinguishes the server definition, and press ENTER. You can list multiple server definitions to be deleted, separating server definition names with the \ character. You are then prompted to provide a name for the response.ini file.
To generate a response.ini file

2 By default, the configuration is saved as Response.ini in the 
\HOME_PATH/ directory, where \HOME_PATH is the directory you 
specified as the Home Directory during installation. You can leave the 
field blank to use the default name or type a different name, and then 
press ENTER. The response.ini file is generated, and you are 
prompted whether to run the configuration immediately.

3 You can choose to run the configuration you just completed or to run the 
configuration using the response.ini file at a different time, as 
described below:

- Type \textbf{Y}, and then press ENTER to run the configuration.
  
  You can also use the response.ini file created for future 
  configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

- Type \textbf{N}, and then press ENTER to quit without running the 
  configuration. You can use the response.ini file created for future 
  configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

Creating a project source

If you selected option 3 in Configuration tasks, page 432, you can create 
project sources, as described in the procedure below.

To create a project source

1 In the prompt that asks for a project source name, type the name for the 
project source to be created, and then press ENTER. You are then 
prompted to provide the Intelligence Server name.

2 Type the Intelligence Server name, and then press ENTER. You can also 
press ENTER without typing any information to accept the default 
Intelligence Server. You are then prompted to provide a TCP port to use 
for Intelligence Server.
3 You can use the default port number or enter a different port number for Intelligence Server, as described below:

- Leave the prompt blank, and then press ENTER to use the default port number.
- Type a port number, and then press ENTER.

You are then prompted to define a time interval for a project source connection time out.

4 Type a numerical value (in minutes) for the amount of inactivity that is allowed before a user is automatically disconnected from a project source. This enforces a connection time out for inactive users connected to a project source. Type 0 to define that users are not disconnected from project sources due to inactivity. Then press ENTER.

You are then prompted to select an authentication type for the project source.

5 You can type the corresponding number to select one of the authentication types listed in the command line. For information on each authentication type, see Authentication modes, page 181.

You are then prompted to provide a name for the response.ini file.

To generate a response.ini file

6 By default, the configuration is saved as Response.ini in the /HOME_PATH/ directory, where HOME_PATH is the directory you specified as the Home Directory during installation. You can leave the field blank to use the default name or type a different name, and then press ENTER. The response.ini file is generated, and you are prompted whether to run the configuration immediately.

7 You can choose to run the configuration you just completed or to run the configuration using the response.ini file at a different time, as described below:

- Type Y, and then press ENTER to run the configuration.

  You can also use the response.ini file created for future configurations, as described in Using the response.ini file to configure MicroStrategy, page 448.

- Type N, and then press ENTER to quit without running the configuration. You can use the response.ini file created for future
configurations, as described in *Using the response.ini file to configure MicroStrategy, page 448.*

**Using the response.ini file to configure MicroStrategy**

This section describes how to use the `response.ini` file through the MicroStrategy Configuration Wizard. For information on how to configure through the MicroStrategy Configuration Wizard, see *Creating a response.ini file, page 431.*

---

**To use the response.ini file through the Configuration Wizard in command line mode**

1. In a UNIX or Linux console window, browse to `HOME_PATH` where `HOME_PATH` is the directory you specified as the Home Directory during installation.

2. Browse to the folder `bin`.

3. Type `mstrcfgwiz-editor -r ResponseFile`, where `ResponseFile` is the full qualified path to the `response.ini` file. For example:

   ```
mstrcfgwiz-editor -r
/home/username/MicroStrategy/RESPONSE.INI
   ```

4. Press ENTER.

   If the path and the response file are valid, the configuration is performed and a successful configuration message appears. If an error occurs before or during the process, an error message displays the error that occurred when executing the response file.

**Parameters and options in the response.ini file**

For a list of all parameters and options available for a response.ini file, see *Response configuration parameters and options, page 186.*
Configuring and controlling Intelligence Server

MicroStrategy provides various command line tools to configure and control Intelligence Servers running on UNIX and Linux. Each command line tool provides descriptive prompts and help information to guide you on how you can use the tool. This section gives a general overview of each tool's functionality, and how to access more detailed information on how to use the tools.

On a Windows machine, these configurations can be completed with MicroStrategy Desktop, Service Manager, and other MicroStrategy tools. For information on performing various administrative tasks, see the System Administration Guide.

Starting, configuring, and monitoring Intelligence Server with mstrsvr

If your Intelligence Server is installed on a UNIX or Linux machine, you can start, configure, and monitor your Intelligence Server from the command line with mstrsvr. This tool starts Intelligence Server from the command line and displays the following information about your Intelligence Server connection:

- Intelligence Server version number
- Intelligence Server instance name
- Metadata DSN
- Metadata login
- Intelligence Server definition name
- Port number

You can then perform various configuration and monitoring tasks for your running Intelligence Server, which includes but is not limited to:

- Display database connection information
- Open, idle, and resume projects
- Check and close jobs
- Monitor users
- Define server clustering options
- Monitor memory usage information
- Stop the server
- Monitor lock contentions

**To start, configure, and monitor Intelligence Server with mstrsvr**

1. From a UNIX/Linux console window, browse to `HOME_PATH`, where `HOME_PATH` is the directory that you specified as the home directory during installation.

2. Browse to the folder `bin`.

3. Type `mstrsvr`, and then press `ENTER` to start Intelligence Server and display available configuration and monitoring options.

4. Once Intelligence Server is started, general configuration information is displayed along with all available configuration and monitoring options. Perform any configuration and monitoring tasks you require.

5. To quit the tool and stop Intelligence Server, type `S`, and then press `ENTER`.

**Configuring the default server instance with mstrsvr-configure**

You can configure the default server instance for Intelligence Server using `mstrsvr-configure`, which is a wizard-style command line tool that prompts you for the required information.

**To configure the default server instance with mstrsvr-configure**

1. From a UNIX/Linux console window, browse to `HOME_PATH`, where `HOME_PATH` is the directory that you specified as the home directory during installation.

2. Browse to the folder `bin`.
3 Type `mstrsvr-configure`, and then press ENTER. You are then prompted to provide a port number for Intelligence Server.

4 Type a port number, and then press ENTER. You are then prompted to provide a DSN to connect to.

5 Type a DSN, and then press ENTER. You are then prompted to provide a valid login for the DSN.

6 Type a valid login for the DSN, and then press ENTER. You are then prompted to provide a password for the DSN login.

7 Type a valid password for the DSN login, and then press ENTER. You are then prompted to provide a server definition name.

8 Type a server definition name, and then press ENTER. Your default server instance is configured.

Creating and configuring Intelligence Server instances with `mstrctl`

You can create and configure Intelligence Server instances with the `mstrctl` tool. Intelligence Servers running with a particular server definition are referred to as server instances.

To create and configure Intelligence Server instances with `mstrctl`

1 From a UNIX/Linux console window, browse to `HOME_PATH`, where `HOME_PATH` is the directory that you specified as the home directory during installation.

2 Browse to the folder `bin`.

3 Type `mstrctl -h`, and then press ENTER. Help information is displayed, which provides syntax standards and available configuration options.

4 Review the help information and run any required configuration tasks.

There are some commands that can output information to a file, or require a long definition that can be retrieved from a file. For information on using files to store output from and provide input to `mstrctl` commands, see *Using files to store output and provide input*, page 452.
You do not need to enter any command to quit the mstrctl tool because it is a one-line command line tool.

**Using files to store output and provide input**

With the mstrctl command line tool, you can perform the following tasks:

- Display and modify a server configuration
- Display and modify a service configuration
- Display and modify a server instance configuration

The commands that display the configurations listed above output long XML definitions to the command line. The commands that modify the configurations listed above require a long XML definition as input.

Rather than displaying and inputting long XML definitions from the command line, you can use files to store and provide input for long XML definitions.

Be aware of the following:

- Configuring Intelligence Server with XML files requires extensive knowledge of the various parameters and values used to define Intelligence Server configurations. Providing an incorrect XML definition to configure Intelligence Server can cause errors and unexpected functionality.

- Prior to using commands to display and modify service configurations (gsvc and ssvc) you must register Intelligence Server as a service. You can perform this task by using the rs command for mstrctl. To register an Intelligence Server as a service on a UNIX or Linux machine, you must be logged in with an account that has root user privileges and permissions.

The following commands can have their output sent to a file:

- **gsc**: Displays a server configuration
- **gsvc**: Displays a service configuration
- **gsic**: Displays a server instance configuration
For example, you can run the following command to output the server instance configuration to an XML file:

```
mstrctl -s IntelligenceServer gsic > ServerInstance.xml
```

A `ServerInstance.xml` file is saved in the current directory.

The following commands can read input from a file:
- `ssc`: Modifies a server configuration
- `ssvc`: Modifies a service configuration
- `ssic`: Modifies a server instance configuration

For example, you can run the following command to modify the server instance configuration by reading input from an XML file:

```
mstrctl -s IntelligenceServer ssic < ServerInstance.xml
```

The XML definition in `ServerInstance.xml` is used to define the server instance configuration.

It would be difficult and time consuming to type a complete server, service, or server instance configuration from the command line. An easier way to provide this type of configuration is to output the current configuration to a file, modify the file with a text editor, and then use the file as input to a command to modify the configuration.

## Supporting reserved words and characters

When you perform MicroStrategy configuration tasks through the UNIX or Linux operating system console, you must make sure reserved words and characters are not mistakenly included in your commands.

UNIX and Linux operating system consoles use reserved words and characters to perform various actions. For example, the `$` character may perform an action when included as part of a command executed through the operating system console. If this character is included in a command to configure, it can cause the command to fail.
For example, you use the following command to create a DSN to an Oracle database:

```
mstrconnectwiz ORCLW $MyOracleDSN 12.34.56.78 orcl 1521
-u:OracleUser -p:OracleUserPassword
```

Notice that the name of the DSN begins with the $ character. If this is a reserved character, the command fails to execute properly.

To avoid this problem, you can place single quotes (‘’) around any character strings that may include reserved words or characters. This prevents the operating system console from interpreting the characters as an operating system action, and instead includes them as part of the character string. For example, the same command as above to create a DSN can be rewritten as follows:

```
mstrconnectwiz ORCLW ‘$MyOracleDSN’ 12.34.56.78 orcl
1521 -u:OracleUser -p:OracleUserPassword
```

This time, the name of the DSN $MyOracleDSN is enclosed by single quotes, which allows all of the characters to be interpreted as a string of characters.

**Configuring your MicroStrategy installation**

To help guide the rest of your installation and configuration steps, refer to the section *Installing and configuring MicroStrategy on UNIX and Linux, page 89* in *Chapter 1, Planning Your Installation*, for an installation and configuration checklist.
13

Adding or Removing MicroStrategy Components

Introduction

This chapter describes how to add or remove MicroStrategy components on different operating systems.

This chapter includes the following sections:

- Adding or removing MicroStrategy components on Windows
- Re-installing MicroStrategy components on Windows
- Uninstalling MicroStrategy components on Windows
- Uninstalling MicroStrategy components on UNIX and Linux
Adding or removing MicroStrategy components on Windows

You can choose to add or remove one or more MicroStrategy components.

If you installed the MicroStrategy components using a disk, you need your original installation disk to add or remove MicroStrategy components.

---

To add or remove MicroStrategy components

2. On the Windows Start menu, point to Settings, and then choose Control Panel. The Control Panel window opens.
3. Double-click Add/Remove Programs. The Add/Remove Programs dialog box opens.
5. Select Modify and click Next.
6. Select to accept the license agreement and click Next.
7. Verify your customer information and click Next.
8. Verify your setup type and click Next.
9. Select the components to add by selecting their check boxes. Clear the check boxes for the components you want to uninstall. Click Next.

The components that are currently installed are displayed with their check boxes selected. These components are not re-installed during the modification process. If you clear any of the check boxes, that particular component is uninstalled during the modification process. You are advised not to clear the check boxes of the components that are already installed, unless you want to remove the component.

10. If you are prompted to stop your Web server, click Yes to stop it and continue with adding or removing files.

© 2011 MicroStrategy, Inc.
11. Verify the settings and click **Next** to begin copying or removing the files.

12. After the modification routine is complete, click **Finish** to close the maintenance program.

For more details on each page of the MicroStrategy Installation Wizard, see *Chapter 2, Installing MicroStrategy on Windows*.

---

**Re-installing MicroStrategy components on Windows**

You can re-install MicroStrategy components if you face problems with previously installed components. During re-installation the list of components previously installed are displayed and these components are re-installed. If you installed the MicroStrategy components using a disk, you need your original installation disk to repair the installation.

The re-installation of MicroStrategy Office must be performed separately. The procedure for re-installing MicroStrategy Office is explained in the following section.

---

**To re-install MicroStrategy components**


2. On the Windows **Start** menu, select **Settings**, and then choose **Control Panel**. The Control Panel window opens.

3. Double-click **Add/Remove Programs**. The Add/Remove Programs dialog box opens.


5. Select **Repair** and click **Next**.

6. Accept the license agreement and click **Next**.

7. You are prompted to select **Yes** to continue with the re-installation procedure and overwrite the components. If you do not want to overwrite the components, select **No**.
8 If you are prompted to stop your Web server, click Yes to stop it and continue with the re-installation.

9 After the re-installation routine is complete, click Finish to close the maintenance program.

For details on each page of the MicroStrategy Installation Wizard, see Chapter 2, Installing MicroStrategy on Windows.

Re-installing MicroStrategy Office

This section describes the re-installation procedure for MicroStrategy Office.

---

**To re-install MicroStrategy Office**

1 Close all installed MicroStrategy products.

2 On the Windows Start menu, select Settings, and then choose Control Panel. The Control Panel window opens.

3 Double-click the Add/Remove Programs icon. The Add/Remove Programs dialog box opens.


5 Select Repair and click Next.

6 Accept the license agreement and click Next.

7 You are prompted to select Yes to continue with the re-installation procedure and overwrite the components. If you do not want to overwrite components, select No.

8 If you are prompted to stop your Web server, click Yes to stop it and continue with the re-installation.

9 After the re-installation routine is complete, click Finish to close the maintenance program.

For details on each page of the MicroStrategy Installation Wizard, see Chapter 2, Installing MicroStrategy on Windows.
Uninstalling MicroStrategy components on Windows

The uninstallation procedure performs the following functions:

- Unregisters and removes selected files, registry entries, and shortcuts logged in the Uninst.isu log file.
- Calls a custom DLL to handle unlogged items, such as registry entries and files.

Before uninstallation begins, the DLL file performs the following functions:

- Checks for user privileges. If they are not valid, uninstallation stops.
- Checks for running components. If a component is found running, uninstallation stops.
- Stops and deletes the MicroStrategy Intelligence Server service. This occurs only when the Intelligence Server is being uninstalled.
- Deletes files created by the application, such as *.log, *.gid, *.ldb and *.tb.

The uninstallation of MicroStrategy Office must be performed separately. The procedure for uninstalling MicroStrategy Office is explained in the following sections.

To uninstall MicroStrategy components on Windows

2. On the Windows Start menu, select **Settings**, and then **Control Panel**. The Control Panel window opens.
3. Double-click **Add/Remove Programs**. The Add/Remove Programs dialog box opens.
5. Select **Remove** and click **Next**.
6. Click **Yes** to any prompts that appear.
7 If you are prompted to stop your Web server, click Yes to stop it and continue with the uninstallation.

8 After the uninstall routine is complete, select Yes to restart your computer, or No to restart it later.

9 Click Finish to close the maintenance program.

You should restart the computer for a clean uninstall.

Uninstalling MicroStrategy Office

This section describes the uninstalling procedure for MicroStrategy Office.

To uninstall MicroStrategy Office

1 Close all installed MicroStrategy products.

2 On the Windows Start menu, select Settings, and then Control Panel. The Control Panel window opens.

3 Double-click Add/Remove Programs. The Add/Remove Programs dialog box opens.


5 Select Remove and click Next.

6 You are prompted to select Yes to continue with the uninstallation procedure. To not remove the application with all of its components, select No.

7 After the uninstallation routine is complete, click Finish to close the maintenance program.
Uninstalling MicroStrategy components on UNIX and Linux

This section discusses how to uninstall MicroStrategy Intelligence Server, Web Universal, and other MicroStrategy products on a UNIX or Linux platform.

To uninstall MicroStrategy Intelligence Server

1. In a UNIX or Linux console window, browse to INSTALL_PATH, where INSTALL_PATH is the directory you specified as the install directory during installation.

2. Browse to the /_uninst folder.

3. You can uninstall MicroStrategy products using the MicroStrategy Installation Wizard or through the command line as described in the options below:

   • To uninstall using the MicroStrategy Installation Wizard, type ./uninstall.bin and press ENTER. Follow the steps described in To complete uninstallation with the MicroStrategy Installation Wizard below to complete the uninstallation process.

   • To uninstall through the command line, type ./uninstall.bin -console and press ENTER. Follow the command line prompts to complete the uninstallation process.

To complete uninstallation with the MicroStrategy Installation Wizard


5. On the Product Uninstallation page, select the products to uninstall. Click Next.

6. On the Summary page, verify the information and click Next. The products are uninstalled.

7. After uninstallation is complete, a message is displayed. Click Finish to complete the process and exit the MicroStrategy Installation Wizard.
Connecting to Databases: ODBC and DSNs

Introduction

This appendix describes the DSN parameters required for ODBC drivers to connect MicroStrategy to various databases. DSNs can be created using the MicroStrategy Connectivity Wizard, and in UNIX and Linux, you can also configure ODBC parameters with the `odbc.ini` file. This appendix discusses these two topics in the following sections:

- *Creating DSNs for specific ODBC drivers, page 463*
- *Configuring ODBC parameters with odbc.ini, page 499*

Creating DSNs for specific ODBC drivers

This section provides information on creating a DSN for each ODBC driver available through the Connectivity Wizard. The following table lists the information required for each type of ODBC driver when you create a new
DSN using the Connectivity Wizard. For information on what operating systems each ODBC driver is certified for, see *Certified ODBC drivers for MicroStrategy Intelligence Server, page 79.*

Note the following:

- For ODBC-specific driver details, refer to the different ODBC driver sections below the table.

- You can create a DSN from the command line in UNIX or Linux. Browse to `HOME_PATH/bin`, where `HOME_PATH` is the directory you specified as the home directory during installation. In the console window, type `./mstrconnectwiz -h`, then press ENTER. This command displays command line syntax and examples for different database platforms. Create your command based on the syntax and examples displayed. Once you run your command, a DSN is created in the `odbc.ini` file.

<table>
<thead>
<tr>
<th>ODBC Driver</th>
<th>Driver Details</th>
</tr>
</thead>
</table>
| **MicroStrategy ODBC Driver for DB2 Wire Protocol for UDB for Windows** | • Data source name  
• IP address  
• TCP port (in most cases, 50000)  
• Database name  
• Default user ID  
• Password |
| **MicroStrategy ODBC Driver for DB2 UDB for UNIX/Linux** | • Data source name  
• Host name  
• Database name  
• Port number |
| **MicroStrategy ODBC Driver for DB2 Wire Protocol for UDB iSeries/DB2 for i for Windows and UNIX/Linux** | • Data source name  
• IP address  
• Collection  
• Location  
• Isolation level  
• Package owner  
• TCP port (446 in most cases) |
| **MicroStrategy ODBC Driver for DB2 z/OS for Windows and UNIX/Linux** | • Data source name  
• IP address  
• Collection  
• Location  
• Package collection  
• Package owner  
• TCP port (446 in most cases) |
| **MicroStrategy ODBC Driver for Greenplum Wire Protocol for Windows and UNIX/Linux, page 469** | • Data source name  
• Host name  
• Port number (5432 in most cases)  
• Database name |
<table>
<thead>
<tr>
<th>ODBC Driver</th>
<th>Driver Details</th>
</tr>
</thead>
</table>
| MicroStrategy ODBC Driver for Informix Wire Protocol for Windows and UNIX/Linux, page 470 | • Data source name  
• Server name  
• Host name  
• Port number (1526 in most cases)  
• Database name |
| MicroStrategy ODBC Driver for Informix 8 for Windows, page 470 | • Data source name  
• Database name  
• Server name  
• Host name  
• Service name  
• Protocol type |
| MicroStrategy ODBC Driver for MySQL Wire Protocol for Windows and Linux, page 471 | • Data source name  
• Host name  
• Database name  
• Port number  
• User name |
| MicroStrategy ODBC Driver for Oracle Wire Protocol for Windows and UNIX/Linux, page 472 | Data source name and either:  
Standard connection:  
• Host name  
• Port number (in most cases, 1521)  
• SID (MicroStrategy default is ORCL)  
• Service name  
• Alternate servers  
TNSNames connection:  
• Server name  
• TNSNames file |
| MicroStrategy ODBC Driver for Sybase ASE Wire Protocol for Windows and UNIX/Linux, page 473 | • Data source name  
• Network address  
• Database name  
• Enable unicode support |
| MicroStrategy ODBC Driver for PostgreSQL Wire Protocol for Windows and UNIX/Linux, page 473 | • Data Source Name  
• Host Name  
• Port Number  
• Database Name  
• User Name |
| SQL Server for Windows and UNIX/Linux, page 474 | • Data source name  
• Server name  
• Database name  
• Use Windows NT Authentication for login ID |
Creating DSNs for specific ODBC drivers

This section also provides information on how to install drivers from other vendors with MicroStrategy. The following ODBC drivers for Windows and UNIX from other vendors are described as well:

- **ODBC Driver for Red Brick for UNIX/Linux, page 480**
- **ODBC Driver for Sybase Adaptive Server IQ for UNIX/Linux, page 482**
- **ODBC Driver for Teradata for UNIX/Linux, page 483**
- **ODBC Driver for Informix 8 for UNIX, page 485**

The ODBC Driver for Informix 8 for UNIX and Linux is a MicroStrategy-branded ODBC driver, but it is not accessible through the Connectivity Wizard.

- **ODBC Driver for Netezza for UNIX/Linux, page 486**
- **ODBC Driver for MySQL Community Server 5.1 for Linux, page 488**
- **ODBC Driver for Aster nCluster for Linux, page 490**
- **ODBC Driver for Vertica for Linux or Solaris, page 492**
- **Other data sources and relational databases for Windows, page 493**

The following sections provide more details about each individual driver’s parameters.
MicroStrategy ODBC Driver for DB2 Wire Protocol for UDB for Windows

The following information is required for setting up the driver connection for MicroStrategy ODBC Driver for DB2 Wire Protocol when running against DB2 in Windows:

- **Data Source Name**: A name to identify the DB2 data source configuration in MicroStrategy. For example, Finance or DB2-1 can serve to identify the connection.
- **IP Address**: The IP Address of the machine where the catalog tables are stored. You can use an IP address such as 123.456.78.90, or a host name such as localhost.
- **TCP Port**: The DB2 DRDA listener process's port number on the server host machine provided by your database administrator. The default port number is usually 50000.
- **Database Name**: The name of the database to connect to by default.
- **Default User ID**: The default user logon ID for your DB2 database.
- **Password**: The password necessary to connect to the DB2 database for the user logon entered in the Default User ID text box above.

Connection to a DB2 UDB for OS/390 database is supported (not certified) for use with MicroStrategy. For information on connecting to a DB2 UDB for OS/390 database, refer to MicroStrategy Tech Note TN11223.

MicroStrategy ODBC Driver for DB2 UDB for UNIX/Linux

The following information is required for setting up the driver connection for MicroStrategy ODBC Driver for DB2 UDB when running against DB2:

- **Data Source Name**: A name to identify the DB2 UDB data source configuration in MicroStrategy. For example, Finance or DB2-Serv1 can serve to identify the connection.
- **Host Name**: The name of the machine that runs the DB2 UDB server.
- **Database Name**: The name of the database to connect to by default, which is assigned by the database administrator.
• **Port Number**: The DB2 UDB server listener's port number. In most cases, the default port number is **50000**, but you should check with your database administrator for the correct number.

**MicroStrategy ODBC Driver for DB2 Wire Protocol for UDB iSeries/DB2 for i for Windows and UNIX/Linux**

The following information is required for setting up the driver connection for MicroStrategy ODBC Driver for DB2 Wire Protocol for UDB iSeries/DB2 for i:

• **Data Source Name**: A name to identify the DB2 for i data source configuration in MicroStrategy. For example, Finance or DB2fori-1 can serve to identify the connection.

• **IP Address**: The IP Address of the machine where the catalog tables are stored. This can be either a numeric address such as **123.456.789.98**, or a host name. If you use a host name, it must be located in the `HOSTS` file of the machine or a DNS server.

• **Collection**: The name that identifies a logical group of database objects.

• **Location**: The DB2 location name, which is defined during the local DB2 installation.

• **Isolation Level**: The method by which locks are acquired and released by the system.

• **Package Owner**: The package's AuthID if you want to specify a fixed user to create and modify the packages on the database. The AuthID must have authority to execute all the SQL in the package.

• **TCP Port**: The DB2 DRDA listener process's port number on the server host machine provided by your database administrator. The default port number is usually **446**.
MicroStrategy ODBC Driver for DB2 z/OS for Windows and UNIX/Linux

The following information is required for setting up the driver connection for MicroStrategy ODBC Driver for DB2 z/OS (formerly known as OS/390):

- **Data Source Name**: A name to identify the DB2 z/OS data source configuration in MicroStrategy. For example, Finance or DB2UDBz/OS-1 can serve to identify the connection.

- **IP Address**: The IP Address of the machine where the catalog tables are stored. This can be either a numeric address such as 123.456.789.98, or a host name. If you use a host name, it must be located in the HOSTS file of the machine or a DNS server.

- **Collection**: The name that identifies a logical group of database objects, which is also the current schema. On DB2 z/OS, the user ID should be used as the Collection.

- **Location**: The DB2 z/OS location name, which is defined during the local DB2 z/OS installation. To determine the DB2 location, you can run the command DISPLAY DDF.

- **Package Collection**: The collection or location name where bind packages are created and stored for searching purposes.

- **Package Owner (Optional)**: The package's AuthID if you want to specify a fixed user to create and modify the packages on the database. The AuthID must have authority to execute all the SQL in the package.

- **TCP Port**: The DB2 DRDA listener process's port number on the server host machine provided by your database administrator. The default port number is usually 446.

MicroStrategy ODBC Driver for Greenplum Wire Protocol for Windows and UNIX/Linux

The following information is required for setting up the driver connection for the MicroStrategy ODBC Driver for Greenplum Wire Protocol:

- **Data Source Name**: A name to identify the Greenplum data source configuration in MicroStrategy. For example, Finance or Greenplum-1 can serve to identify the connection.
• **Host Name**: The name or IP address of the machine on which the Greenplum data source resides. The system administrator or database administrator assigns the host name.

• **Port Number**: The port number for the connection. The default port number for Greenplum is usually 5432. Check with your database administrator for the correct number.

• **Database Name**: The name of the database to connect to by default. The database administrator assigns the database name.

MicroStrategy ODBC Driver for Informix Wire Protocol for Windows and UNIX/Linux

The following information is required for setting up the driver connection for the MicroStrategy ODBC Driver for Informix Wire Protocol:

• **Data Source Name**: A name to identify the Informix data source configuration in MicroStrategy. For example, Finance or Informix-1 can serve to identify the connection.

• **Server Name**: The client connection string designating the server and database to be accessed.

• **Host Name**: The name of the machine on which the Informix server resides. The system administrator or database administrator assigns the host name.

• **Port Number**: The Informix server listener's port number. The default port number for Informix is commonly 1526.

• **Database Name**: The name of the database to connect to by default, which is assigned by the database administrator.

MicroStrategy ODBC Driver for Informix 8 for Windows

Informix Client Software (Client-SDK) must be installed before you create a DSN. However, this software is not included in the MicroStrategy product suite installation. It must be obtained through the database vendor or a third party. For more information, see *ODBC Driver for Informix 8 for UNIX, page 485*. MicroStrategy does not support this platform for use as a metadata repository.
The following information is required for setting up the driver connection for the MicroStrategy ODBC Driver for Informix 8:

- **Data Source Name**: A name to identify the Informix data source configuration in MicroStrategy. For example, Finance or Informix-1 can serve to identify the connection.

- **Database**: The name of the database to connect to by default, which is assigned by the database administrator.

- **Server Name**: The client connection string designating the server and database to be accessed.

- **Host Name**: The name of the machine on which the Informix server resides.

- **Service Name**: The service name, as it exists on the host machine. The system administrator assigns the service name.

- **Protocol Type**: The protocol used to communicate with the server.

### MicroStrategy ODBC Driver for MySQL Wire Protocol for Windows and Linux

The MicroStrategy ODBC Driver for MySQL Wire Protocol is for use with MySQL Enterprise 5.0 on Windows and Linux. The following information is required for setting up the driver connection for the MicroStrategy ODBC driver for MySQL Wire Protocol:

- **Data Source Name**: A name to identify the MySQL data source configuration in MicroStrategy. For example, Finance or MySQL-1 can serve to identify the connection.

- **Host Name**: The name or IP address of the machine on which MySQL Enterprise 5.0 resides. The system administrator or database administrator assigns the host name.

- **Database Name**: The name of the database to connect to by default. The database administrator assigns the database name.

- **Port Number**: The port number for the connection. The default port number for MySQL is usually 3306. Check with your database administrator for the correct number.

- **User Name**: The name of a valid user for MySQL Enterprise 5.0.
MicroStrategy ODBC Driver for Oracle Wire Protocol for Windows and UNIX/Linux

The following information is required for setting up the driver connection for MicroStrategy ODBC driver for Oracle Wire Protocol:

**Data Source Name**: Enter a name to identify the Oracle data source configuration in MicroStrategy. For example, Finance or Oracle-1 can serve to identify the connection. A DSN is required for any Oracle Wire Protocol connection. Depending on whether you want to use a standard connection or a TNSNames connection, refer to one of the following lists of options below:

- **Standard Connection**: A standard connection is configured through Oracle Wire Protocol with the following connection parameters:
  - **Host Name**: The name of the Oracle server to be accessed. This can be a server name such as Oracle-1 or an IP address such as 123.456.789.98.
  - **Port Number**: The Oracle listener port number provided by your database administrator. The default port number is usually 1521.
  - One of the following parameters; which one you choose is up to your personal preference:
    - **SID**: The Oracle System Identifier for the instance of Oracle running on the server. The default SID is usually ORCL.
    - **Service Name**: The global database name, which includes the database name and the domain name. For example, if your database name is finance and its domain is business.com the service name is finance.business.com.
  - **Alternate Servers**: A list of alternate database servers to enable connection failover for the driver. If the primary database server entered as the SID or service name is unavailable, a connection to the servers in this list is attempted until a connection can be established. You can list the servers in SID or service name format, as shown in the following examples:
    - Using an SID: (HostName=DB_server_name:PortNumber=1526:SID=ORCL)
    - Using a Service Name: (HostName=DB_server_name:PortNumber=1526:ServiceName=service.name.com)

- **TNSNames Connection**: A TNSNames connection uses a TNSNAMES.ORA file to retrieve host, port number, and SID information.
from a server (alias or Oracle net service name) listed in the TNSNAMES.ORA file. A TNSNames connection requires the following parameters:

- **Server Name**: A server name, which is included in a TNSNAMES.ORA file included in the TNSNames File text box below.

- **TNSNames File**: The location of your TNSNAMES.ORA file. Make sure to enter the entire path to the TNSNAMES.ORA file, including the file name itself. You can specify multiple TNSNAMES.ORA files.

### MicroStrategy ODBC Driver for Sybase ASE Wire Protocol for Windows and UNIX/Linux

The following information is required for setting up the driver connection for the MicroStrategy ODBC driver for Sybase ASE Wire Protocol:

- **Data Source Name**: A name to identify the Sybase ASE data source configuration in MicroStrategy. For example, Finance or SybaseASE-1 can serve to identify the connection.

- **Network Address**: The network address, in the format `ServerName_or_IPAddress,PortNumber`. For example, if your network supports named servers, you can specify an address such as `SybaseASE-1,5000`. You can also specify the IP address such as `123.456.789.98,5000`. Contact your system administrator for the server name or IP address.

- **Database Name**: The name of the database to connect to by default. The database administrator assigns the database name.

- **Enable Unicode support (UTF8)**: Select this check box if the database supports unicode.

### MicroStrategy ODBC Driver for PostgreSQL Wire Protocol for Windows and UNIX/Linux

The following information is required for setting up the driver connection for the MicroStrategy ODBC driver for PostgreSQL Wire Protocol:

- **Data Source Name**: A name to identify the PostgreSQL data source configuration in MicroStrategy. For example, Finance or PostgreSQL-1 can serve to identify the connection.
- **Host Name**: The name or IP address of the machine on which the PostgreSQL database resides. The system administrator or database administrator assigns the host name.

- **Port Number**: The port number for the connection. The default port number for PostgreSQL is usually **5432**. Check with your database administrator for the correct number.

- **Database Name**: The name of the database to connect to by default. The database administrator assigns the database name.

- **User Name**: The name of a valid user for the PostgreSQL database.

### SQL Server for Windows and UNIX/Linux

Microsoft SQL Server is used for Windows platforms. However, a MicroStrategy-branded version of SQL Server is used for UNIX and Linux.

The following information is required for setting up the driver connection for the Microsoft SQL Server driver and the MicroStrategy-branded version of the SQL Server driver:

- **Data Source Name**: A name to identify the Microsoft SQL Server data source configuration in MicroStrategy. For example, Personnel or SQLServer-1 can serve to identify the connection.

- **Server Name**: The name of a SQL Server on your network, in the format `ServerName_or_IPAddress,PortNumber`. For example, if your network supports named servers, you can specify an address such as `SQLServer-1,1433`. You can also specify the IP address such as `123.45.678.998,1433`.

- **Database Name**: The name of the database to connect to by default. The database administrator assigns the database name.

- **Use Windows NT Authentication for login ID**: This option is available if you are configuring the Microsoft SQL Server driver on Windows. Select this check box to use Windows NT authentication to pass a user’s credentials on the Windows machine to execute against a SQL Server database.

If you use Windows NT authentication with SQL Server, you must enter the Windows NT account user name and password in the Service...
Manager. For information on the Service Manager, see the System Administration Guide.

Inserting date data into SQL Server 2000 tables can cause errors if the system’s Regional Settings are not set properly. Ensure that the date format is defined to be in an English format.

**Netezza SQL for Windows**

The following information is required for setting up the driver connection for the Netezza SQL driver in Windows:

- **Data Source Name**: The name used for the data source connection request.
- **Server Name**: The machine name or IP address of the machine where the database is stored.
- **Database Name**: The name of the database to connect with the Netezza SQL driver.
- **Port Number**: The port number for the connection. Check with your database administrator for the correct number.
- **User Name**: The name of a valid user for the database.
- **Password**: The password for the user name you provided to connect to the database.

**MicroStrategy ODBC Driver for SequeLink**

The MicroStrategy ODBC Driver for SequeLink allows you to access Microsoft Access databases or Microsoft Excel files stored on a Windows machine from an Intelligence Server hosted on a UNIX or Linux machine. The steps below show you how to perform the necessary configurations on the various machines to support this type of configuration:

- *Preparing the Microsoft Access database, page 476*
- *Preparing the Microsoft Excel file, page 477*
- *Configuring the MicroStrategy ODBC driver for SequeLink, page 479*
Preparing the Microsoft Access database

You must complete the steps provided below to access an Access database stored on a Windows machine from an Intelligence server hosted on a UNIX or Linux machine.

Prerequisites

- On the Windows machine where the Access database is stored, you must create a DSN to connect to the Access database. For instructions on creating a DSN, see Creating a DSN for a data source, page 151.

To access Microsoft Access databases from an Intelligence Server hosted on UNIX or Linux

1. On the Windows machine that stores the Access database to connect to, install the SequeLink ODBC Socket Server. This can be installed as part of a MicroStrategy installation, and is included in the Other components options of the MicroStrategy Product Suite (see Select Features, page 96).

2. To configure the SequeLink ODBC Socket Server

   a. On the Windows machine where you installed the SequeLink ODBC Socket Server, from the Start menu, point to Programs, point to DataDirect SequeLink 5.5 Service for ODBC Socket, and then select SequeLink Management Console Snap-in.

   b. Under Console Root, expand SequeLink 5.5 Manager, expand Connected to SLAgent55, expand SequeLink Services, expand SLSocket55, expand Configuration, and then select Data Source Settings.

   c. From the Action menu, point to New, and select Data Source. A new data source is created underneath Data Source Settings.

   d. Type a descriptive name for the new data source, such as Access Data Source.

   e. Expand the new data source and select Advanced.

   f. Right-click DataSourceSOCODBCConnStr and select Properties. The DataSourceSOCODBCConnStr Properties dialog box opens.
In the Value field, type `DSN=AccessDSN`, where `AccessDSN` is the DSN you created to connect to your Access database. This is different from the data source you created as part of the steps to configure the SequeLink ODBC Socket Server.

Click **OK**.

Within the same data source, select **User Security**.

Right click **DataSourceLogonMethod** and select **Properties**. The **DataSourceLogonMethod Properties** dialog box opens.

From the **Value** drop-down list, select **Anonymous**. This allows connection to the Access database without using a user name and password.

Click **OK**.

Right-click the data source, point to **All Tasks**, and select **Save configuration**.

On the UNIX or Linux machine that hosts your Intelligence Server, you must configure the MicroStrategy ODBC driver for SequeLink to connect to the Access database. For instructions on how to perform this configuration, see *Configuring the MicroStrategy ODBC driver for SequeLink, page 479*.

### Preparing the Microsoft Excel file

You must complete the steps provided below to access Excel files stored on a Windows machine from an Intelligence server hosted on a UNIX or Linux machine.

**Prerequisites**

- On the Windows machine where the Excel file is stored, you must prepare the Excel file as a valid data source. For instructions to prepare an Excel file, see *Prepare an Excel file as a valid data source, page 494*.
- On the Windows machine where the Excel file is stored, you must create a DSN to connect to the Excel file. For instructions to create a DSN for an Excel file, see *Create a data source name for an Excel file, page 495*. 
To access Microsoft Excel files from an Intelligence Server hosted on UNIX or Linux

1. On the Windows machine that stores the Excel files to connect to, install the SequeLink ODBC Socket Server. This can be installed as part of a MicroStrategy installation, and is included in the Other components options of the MicroStrategy Product Suite (see Select Features, page 96).

To configure the SequeLink ODBC Socket Server

2. On the Windows machine where you installed the SequeLink ODBC Socket Server, from the Start menu, point to Programs, point to DataDirect SequeLink 5.5 Service for ODBC Socket, and then select SequeLink Management Console Snap-in.

3. Under Console Root, expand SequeLink 5.5 Manager, expand Connected to SLAgent55, expand SequeLink Services, expand SLSocket55, expand Configuration, and then select Data Source Settings.

4. From the Action menu, point to New, and select Data Source. A new data source is created underneath Data Source Settings.

5. Type a descriptive name for the new data source, such as Excel Data Source.

6. Expand the new data source, and select Advanced.

7. Right-click DataSourceSOCODBCConnStr and select Properties. The DataSourceSOCODBCConnStr Properties dialog box opens.

8. In the Value field, type DSN=ExcelDSN, where ExcelDSN is the DSN you created to connect to your Excel file. This is different from that data source you created as part of the steps to configure the SequeLink ODBC Socket Server.

9. Click OK.

10. Within the same data source, select User Security.

12 From the **Value** drop-down list, select **Anonymous**. This allows connection to the Excel file without using a user name and password.

13 Click **OK**.

14 Right-click the data source, point to **All Tasks**, and select **Save configuration**.

15 On the UNIX or Linux machine that hosts your Intelligence Server, you must configure the MicroStrategy ODBC driver for SequeLink to connect to the Excel files. For instructions on how to perform this configuration, see **Configuring the MicroStrategy ODBC driver for SequeLink**, page 479.

**Configuring the MicroStrategy ODBC driver for SequeLink**

The steps provided below show you how to configure the MicroStrategy ODBC driver for SequeLink to access either Microsoft Access databases or Excel files stored on a Windows machine.

---

**To configure the MicroStrategy ODBC driver for SequeLink**

1 On the UNIX or Linux machine that hosts Intelligence Server, browse to `HOME_PATH` where `HOME_PATH` is the directory you specified as the Home Directory during installation.

2 Open the `odbc.ini.example` file and find the section that starts with `[SequeLinkODBC]`. **Copy this section into the `odbc.ini` file.**

3 Edit the parameters listed below:

   - **Host**: Type the IP address of the Windows machine that stores the Access database or Excel files.

   - **ServerDataSource**: Type the name of the data source for the Access database or Excel file to connect to as a data source. This is the name
of the data source that you defined while configuring the SequeLink ODBC Socket Server, as shown in the example image below:

4  Save the `odbc.ini` file.

5  Restart Intelligence Server.

**ODBC Driver for Red Brick for UNIX/Linux**

ODBC driver for Red Brick is not a MicroStrategy-branded driver. The following steps show how to configure ODBC driver for Red Brick.

---

**To create an ODBC Driver for Red Brick**

1  Install ODBC Driver for Red Brick for the correct operating system. For information on installation, refer to the *Installation and Configuration Guide for UNIX and Linux* provided by IBM.

   Make sure you install the Red Brick Client Products (version 6.2 and higher) so that they can be accessed by the appropriate users. You need the following components:

   - RISQL Entry Tool, RISQL Reporter, and Client TMU
   - Red Brick ODBCLib (SDK)
• Red Brick ODBC Driver

The directory where Red Brick Client Products are installed should always be accessible to MicroStrategy Intelligence Server.

**Configure the environment for ODBC Driver for Red Brick**

2 In a UNIX/Linux console window, browse to `HOME_PATH`, where `HOME_PATH` is the directory you specified as the Home Directory during installation. Browse to the folder `env`.

3 Add Write privileges to the ODBC.sh file by entering the following command:

    chmod u+w ODBC.sh

4 Edit the `ODBC.sh` file and add the location of the directory where the Red Brick Client Products are installed (`RED_BRICK_INSTALL_PATH`) to the `RB_CONFIG` environment variable:

    RB_CONFIG=RED_BRICK_INSTALL_PATH

5 Save the `ODBC.sh` file and remove Write privileges from the file by entering the following command:

    chmod a-w ODBC.sh

**Configure a DSN for ODBC Driver for Red Brick**

6 In a UNIX/Linux console window, browse to `HOME_PATH` where `HOME_PATH` is the directory you specified as the Home Directory during installation.

7 Open the `odbc.ini.example` file and find the section that starts with `[RED_BRICK_62]` if you are using Red Brick 6.2 or `[RED_BRICK_63]` if you are using Red Brick 6.3. Copy the section into the `odbc.ini` file.

8 Edit the DSN parameters `SERVER` and `DATABASE`, and modify the value of `RB_CONFIG` with the location of the directory where the Red Brick Client Products are installed.

9 Save the `odbc.ini` file.

For details on these DSN parameters, refer to the product documentation provided directly by the database vendor.
ODBC Driver for Sybase Adaptive Server IQ for UNIX/Linux

ODBC driver for Sybase Adaptive Server IQ is not a MicroStrategy-branded driver. The following steps show how to configure ODBC driver for Sybase Adaptive Server IQ.

To configure ODBC Driver for Sybase Adaptive Server IQ

1. Install ODBC Driver for Sybase Adaptive Server IQ for the correct operating system. For information on installation, refer to the *Installation and Configuration Guide* provided by Sybase.

   The directory where ODBC driver for Sybase Adaptive Server IQ is installed should always be accessible to MicroStrategy Intelligence Server.

   **Configure the environment for ODBC Driver for Sybase Adaptive Server IQ.**

2. In a UNIX/Linux console window, browse to $HOME_PATH$, where $HOME_PATH$ is the directory you specified as the Home Directory during installation. Browse to the folder env.

3. Add Write privileges to the ODBC.sh file by entering the following command:

   ```
   chmod u+w ODBC.sh
   ```

4. Edit the ODBC.sh file and add the location of the directory where the ODBC Driver for Sybase Adaptive Server IQ is installed ($SYBASE_ASIQ_INSTALL_PATH$) to the ASDIR environment variable:

   ```
   ASDIR=$SYBASE_ASIQ_INSTALL_PATH
   ```

5. Save the ODBC.sh file and remove Write privileges from the file by entering the following command:

   ```
   chmod a-w ODBC.sh
   ```

   **Configure the DSN for ODBC Driver for Sybase Adaptive Server IQ.**

6. In a UNIX/Linux console window, browse to $HOME_PATH$ where $HOME_PATH$ is the directory you specified as the home directory during installation.
7 Open the odbc.ini.example file and find the section that starts with [SYBASEIQ]. Copy the section into the odbc.ini file.

8 Edit the DSN parameters EngineName, DatabaseName and CommLinks, and modify the value of ASDIR with the location of the directory where the ODBC Driver for Sybase Adaptive Server IQ is installed.

9 Save the odbc.ini file.

For details on these DSN parameters, refer to the product documentation provided directly by the database vendor.

**ODBC Driver for Teradata for UNIX/Linux**

ODBC driver for Teradata is not a MicroStrategy-branded driver. The following steps show how to configure the ODBC driver for Teradata.

For information on setting up an ODBC driver for Teradata through the Connectivity Wizard on Windows, see *Other data sources and relational databases for Windows, page 493.*

---

**To configure the ODBC Driver for Teradata**

1 Install the ODBC Driver for Teradata for the correct operating system. For information on installation, refer to the product documentation provided directly by the database vendor.

![Warning]
The directory where the ODBC driver for Teradata is installed should always be accessible to MicroStrategy Intelligence Server.

**Specify the Teradata GSS library installation (AIX only)**

2 For AIX only, you need to specify the Teradata GSS library installation in the ODBC.sh. In a UNIX/Linux console window, browse to HOME_PATH where HOME_PATH is the directory you specified as the Home Directory during installation. Browse to the folder env.

3 Add Write privileges to the ODBC.sh file by entering the following command:

```bash
chmod u+w ODBC.sh
```
4 Make the following changes to the MicroStrategy Teradata GSS path. Set this path, in the MSTR_TERADATA_GSS_PATH parameter, to the location of the local Teradata GSS library installation:

```
MSTR_TERADATA_GSS_PATH=<MSTR_TERADATA_GSS_PATH>

if [ "${MSTR_TERADATA_GSS_PATH}" != '<MSTR_TERADATA_GSS_PATH>' ];
then mstr_append_path LIBPATH
"${MSTR_TERADATA_GSS_PATH:?}/lib"
export LIBPATH
```

5 Save the ODBC.sh file and remove Write privileges from the file by entering the following command:

```
chmod a-w ODBC.sh
```

**Configure the DSN for ODBC driver for Teradata**

6 In a UNIX/Linux console window, browse to HOME_PATH where HOME_PATH is the directory you specified as the Home Directory during installation.

7 Open the odbc.ini.example file and find the section that starts with [TERADATA_SERVER]. Copy that section into the odbc.ini file in the [ODBC Data Sources] section.

8 Edit the DSN parameters DBCName, DatabaseName and DefaultDatabase, and modify the value of MSTR_TERADATA_PATH with the location of the directory where the ODBC Driver for Teradata is installed.

You can also edit the parameters in the odbc.ini file to reflect your environment. To support parameterized queries, define the EnableExtendedStmtInfo parameter as EnableExtendedStmtInfo=Yes. This parameter and other Teradata parameters in the odbc.ini file are described in Configuring ODBC parameters with odbc.ini, page 499 and Teradata Server, page 520.

9 Save the odbc.ini file.
ODBC Driver for Informix 8 for UNIX

The MicroStrategy ODBC Driver for Informix 8 is already installed in the INSTALL_PATH/lib32 directory.

However, the Informix Client Software Developer’s Kit (CSDK) must be installed before you create a DSN. This software is not included in the MicroStrategy product suite installation and must be obtained through the database vendor or a third party. For information on installation, refer to the product documentation provided directly by the database vendor.

The following steps show how to configure the MicroStrategy ODBC driver for Informix 8.

To configure ODBC Driver for Informix 8

1. Install the Informix CSDK.

   The directory where CSDK is installed should always be accessible to Intelligence Server.

   Configure the environment for ODBC Driver for Informix 8

2. In a UNIX console window, browse to HOME_PATH, where HOME_PATH is the directory you specified as the Home Directory during installation. Browse to the folder env.

3. Add Write privileges to the ODBC.sh file by entering the following command:

   chmod u+w ODBC.sh

4. Edit the ODBC.sh file and make the following changes:

   - Add the location of the directory where the Informix Client Software Developer's Kit (CSDK) is installed to (CSDK_INSTALL_PATH) the INFORMIXDIR environment variable:

     INFORMIXDIR=CSDK_INSTALL_PATH
• Add the name of the Informix Server \( \text{INFORMIX_SERVER_NAME} \) to the \text{INFORMIXSERVER} environment variable:

\[
\text{INFORMIXSERVER=INFORMIX_SERVER_NAME}
\]

This value is chosen from the list in \(<\text{INFORMIXDIR}>/\text{etc/}
\text{sqlhosts}\).

5 Save the \text{ODBC.sh} file and remove Write privileges from the file by entering the following command:

\[
\text{chmod a-w ODBC.sh}
\]

Configure the DSN for ODBC Driver for Informix 8

6 In a UNIX console window, browse to \text{HOME\_PATH} where \text{HOME\_PATH} is the directory you specified as the home directory during installation.

7 Open the \text{odbc.ini.example} file and search for the section that starts with \[\text{INFORMIX\_XPS}\]. Copy the section into the \text{odbc.ini} file.

8 Edit the DSN parameters Database, HostName, ServerName and Service.

9 Save the \text{odbc.ini} file.

For details on these DSN parameters, refer to the product documentation provided directly by the database vendor.

ODBC Driver for Netezza for UNIX/Linux

ODBC driver for Netezza is not a MicroStrategy-branded driver. The following steps show how to configure ODBC driver for Netezza 4.x and 5.x.

You must modify \text{odbcinst.ini} file and \text{odbc.ini} file to create the DSN for Netezza.
To configure ODBC driver for Netezza

1 Install the ODBC Driver for Netezza for the correct operating system. For information on installation, refer to the product documentation provided directly by the database vendor.

   The directory where Netezza is installed should always be accessible to MicroStrategy Intelligence Server.

2 Configure the DSN for ODBC driver for Netezza.

To modify the odbcinst.ini file

3 In a UNIX/Linux console window, browse to HOME_PATH, where HOME_PATH is the directory you specified as the home directory during installation.

4 Edit the odbcinst.ini file and replace the three instances of netezza_odbc_install_dir with the location of the directory where the Netezza ODBC Driver is installed. An example of this is as follows:

   If the original path is:
   
   Driver = /netezza_odbc_install_dir/lib/
   libnzsqlodbc3.so

   Then the modified path will be:

   Driver = /usr/odbc/netezzahome/lib/libnzsqlodbc3.so

5 Save the odbcinst.ini file.

To modify the odbc.ini file

6 Open the odbc.ini.example file to access the example DSN provided in this file:

   • For Netezza 4.x: Search for the section that starts with [NETEZZA4].
   • For Netezza 5.x: Search for the section that starts with [NETEZZA5].

7 Open the MicroStrategy odbc.ini file.

8 Copy and paste the contents from the odbc.ini.example file for your Netezza ODBC driver. You should paste the contents of the DSN exactly as they appear in the example file.
9 Make the following changes to the copied sample file:

- Modify the driver location to match the location of the installed Netezza ODBC Driver, `netezza_odbc_install_dir`.
- Change the database, server name, user name, and password, and any other relevant parameters to match the information for your database.

10 Save the `odbc.ini` file.

For details on these DSN parameters, refer to the product documentation provided by the database vendor.

**ODBC Driver for MySQL Community Server 5.1 for Linux**

The ODBC driver for MySQL Community Server 5.1 is not a MicroStrategy-branded driver. The following steps show how to configure the ODBC driver for MySQL Community Server 5.1, which is certified for the Linux operating system.

You must modify the `odbc.ini` file to create the DSN for MySQL Community Server 5.1.

The third-party product(s) discussed in the procedure below is manufactured by vendors independent of MicroStrategy. MicroStrategy makes no warranty, express, implied or otherwise, regarding this product, including its performance or reliability.

**To configure ODBC driver for MySQL Community Server 5.1**

1 Install the 32-bit ODBC Driver for MySQL (3.51) for the Linux operating system, found at the hyperlink [http://dev.mysql.com/downloads/connector/odbc/3.51.html](http://dev.mysql.com/downloads/connector/odbc/3.51.html). This site is valid as of the release of this manual. For information on installation, refer to the product documentation provided by the database vendor.

Note the following:

- Ensure that the driver files are installed to the `/usr/lib` directory.
For exact version numbers of MySQL drivers certified with MicroStrategy, refer to the MicroStrategy General Information Readme.

2 Configure the DSN for the ODBC driver for MySQL.

   **To modify the odbc.ini file**

3 In a Linux console window, browse to `HOME_PATH`, where `HOME_PATH` is the directory you specified as the home directory during installation.

4 Open the `odbc.ini.example` file and search for the section that starts with `[MYSQL]`.

5 Open the MicroStrategy `odbc.ini` file.

6 Copy and paste the contents from the `odbc.ini.example` file for your MySQL ODBC driver. You should paste the contents of the DSN exactly as they appear in the example file.

7 Make the following changes to the copied sample file:
   
   - Modify the driver location to match the location of the installed MySQL ODBC Driver.
   - Change the database, server name, user name, password, and any other relevant parameters to match the information for your database.

   Ensure that there is no white space between the equals sign (=) which separates the parameter and its value.

8 Save the `odbc.ini` file.

   You can test a connection to your MySQL database with the MicroStrategy DB Query Tool.

This completes the steps to create a DSN and configure an ODBC driver for MySQL Community Server 5.1. To create a database instance and database connection, see *Creating a database instance, page 197* and *Creating a database connection, page 200*. 
ODBC Driver for Aster nCluster for Linux

The ODBC driver for Aster nCluster is not a MicroStrategy-branded driver. The following steps show how to configure the ODBC driver for nCluster 3.0 for Linux.

You must modify the odbc.ini file to create the DSN for Aster.

⚠️ The third-party product(s) discussed in the procedure below is manufactured by vendors independent of MicroStrategy. MicroStrategy makes no warranty, express, implied or otherwise, regarding this product, including its performance or reliability.

---

To configure ODBC driver for nCluster

1. Install the ODBC Driver for nCluster for the Linux operating system. For information on installation, refer to the product documentation provided by the database vendor.

   Note the following:
   - The path to the installation location you choose for the ODBC driver is used later in this procedure as the value for the Driver parameter in the odbc.ini file.
   - For exact version numbers of Aster drivers certified with MicroStrategy, refer to the MicroStrategy General Information Readme.

2. Configure the DSN for the ODBC driver for nCluster.

   **Configure the environment for ODBC Driver for nCluster**

3. In a Linux console window, browse to HOME_PATH, where HOME_PATH is the directory you specified as the Home Directory during installation. Browse to the folder env.

4. Add Write privileges to the ODBC.sh file by entering the following command:

   chmod u+w ODBC.sh
5 Specify the Aster library installation in the `ODBC.sh` file. To do this, make the following changes to the MicroStrategy Aster path. Set this path, in the `MSTR_TERADATA_GSS_PATH` parameter, to the location of the local Aster library installation:

```bash
# ODBC Driver for Aster nCluster

ASTER_PATH='<ASTER_PATH>'

if [ !"${ASTER_PATH}" != '"<ASTER_PATH>"' ]; then
    export ASTER_PATH
    mstr_append_path LD_LIBRARY_PATH "${ASTER_PATH:?}"
    export LD_LIBRARY_PATH
fi
```

6 Save the `ODBC.sh` file and remove Write privileges from the file by entering the following command:

```bash
chmod a-w ODBC.sh
```

**To modify the `odbc.ini` file**

7 In a Linux console window, browse to `HOME_PATH`, where `HOME_PATH` is the directory you specified as the home directory during installation.

8 Open the `odbc.ini.example` file and search for the section that starts with `[Aster nCluster]`.

9 Open the MicroStrategy `odbc.ini` file.

10 Copy and paste the contents from the `odbc.ini.example` file for your Aster ODBC driver. You should paste the contents of the DSN exactly as they appear in the example file.

11 Make the following changes to the copied sample file:

- Modify the driver location to match the location of the installed Aster ODBC Driver.
- Change the database, server name, user name, password, and any other relevant parameters to match the information for your database.

   Ensure that there is no white space between the equals sign (=) which separates the parameter and its value.
12 Save the odbc.ini file.

This completes the steps to create a DSN and configure an ODBC driver for nCluster.

**ODBC Driver for Vertica for Linux or Solaris**

The ODBC driver for Vertica is not a MicroStrategy-branded driver. The following steps show how to configure the ODBC driver for Vertica for Linux or Solaris.

**To configure ODBC driver for Vertica**

1. Install the ODBC Driver for Vertica for the Linux or Solaris operating system. For information on installation, refer to the product documentation provided by the database vendor.

   Note the following:
   - The path to the installation location you choose for the ODBC driver is used later in this procedure as the value for the Driver parameter in the odbc.ini file.
   - For exact version numbers of Vertica drivers certified with MicroStrategy, refer to the MicroStrategy General Information Readme.

2. Configure the DSN for the ODBC driver for Vertica.

   To modify the odbc.ini file

3. In a Linux or Solaris console window, browse to HOME_PATH, where HOME_PATH is the directory you specified as the home directory during installation.
4 Open the `odbc.ini.example` file and search for the section that starts with `[Vertica]`.

5 Open the MicroStrategy `odbc.ini` file.

6 Copy and paste the contents from the `odbc.ini.example` file for your Vertica ODBC driver. You should paste the contents of the DSN exactly as they appear in the example file.

7 Make the following changes to the copied sample file:
   - Modify the driver location to match the location of the installed Vertica ODBC Driver.
   - Change the database, server name, user name, password, and any other relevant parameters to match the information for your database.
   - Ensure that there is no white space between the equals sign (=) which separates the parameter and its value.

8 Save the `odbc.ini` file.

This completes the steps to create a DSN and configure an ODBC driver for Vertica.

**Other data sources and relational databases for Windows**

If you use other databases or data sources, refer to the database-specific documentation for information on required settings. Standard settings are supported by MicroStrategy for most relational databases.

**Teradata**

If you intend to use Teradata, which is certified by MicroStrategy, you need to:

- Pre-install the NCR ODBC Driver for Teradata RDBMS. Otherwise, you will not see Teradata in the drivers list for Other Relational Databases when using the MicroStrategy ODBC DSN Creator.
- Ensure that Teradata DSNs are set to Run in Quiet Mode.
If you use Teradata, the following settings are required for setting up the driver connection.

In the Teradata ODBC Driver Options dialog box, click **Options** to set the following required options:

- **Session Mode**: Select Teradata as the session mode to apply for the duration of the session.
- **Date Time Format**: Set this value to AAA format so the ODBC driver handles the Dates, Time, and Timestamps as ANSI-compatible strings. The ANSI-compatible strings are only available with the V2R3 or later databases.
- **Disable Parsing**: Select this check box to disable parsing of SQL statements by the ODBC driver.

For information on other options, refer to the online help by clicking **Help**.

**Microsoft Excel**

A Microsoft Excel file can be used as a data source in MicroStrategy. The information provided below explains how to prepare an Excel file for use with MicroStrategy and how to connect to the Excel file.

This data can be used as part of a MicroStrategy project in various ways. For example, you can integrate the Excel data in your project using tools such as Architect, as described in the *Project Design Guide*. You can also use Freeform SQL and Query Builder to access your Excel data, as described in the *Advanced Reporting Guide*.

**Prepare an Excel file as a valid data source**

To use an Excel file as a data source, you must create and store the data in the Excel file so that it can be recognized in MicroStrategy as a set of tables that contain valid data.
To create a table with valid data in an Excel file

1 Prepare the Excel file as follows:
   • Ensure that all column headers are of a valid format:
     – No spaces in the header name (for example, Category_ID instead of Category ID).
     – Alphanumeric, and beginning with a letter.
   • Ensure that all cells for the ID column have a value in them.
2 In the Excel file, create a table by performing the following:
   a Highlight the specific rows and columns with the data to use to create a report with, including the column headers, such as Category_ID and Category_DESC.
      Do not use the column headings at the top of the Excel spreadsheet, marked as A, B, C, and so on to select the whole column. Doing so may include numerous empty cells with NULL values.
   b In the Name Box, type a name for the highlighted cells, and then press ENTER. The name you type in is used in MicroStrategy as a table name.
      The Name Box is the drop-down list on the left-hand side below the toolbars.
      You can create multiple tables in one Excel file by highlighting different parts of the file and assigning them different names.
3 Save the Excel file.
   Ensure that the file is not password-protected.

Create a data source name for an Excel file

To use an Excel file as a data source, you must create a data source name (DSN) for the Excel file. This DSN is used in MicroStrategy to connect to the Excel file.
To create a DSN for an Excel file

1. From the Start menu, point to Programs, then MicroStrategy, then Tools, and then select Connectivity Wizard. The Welcome page of the Connectivity Wizard opens.

2. Read the Welcome page information and click Next. The Driver Selection page opens.

3. Select Other Relational Databases and click Next. A list of additional drivers is displayed.

4. Select the Microsoft Excel Driver and click Next. The ODBC Excel Setup dialog box is displayed.

5. Type a Data Source Name (DSN) for the Excel data source in the space provided.

6. Click Select Workbook. The Select Workbook dialog box is displayed.

7. Browse to and select the Excel file that you saved and named in the previous procedure To create a table with valid data in an Excel file, page 495.

8. Click OK to close the Select Workbook dialog box.

9. Click OK in the ODBC Excel Setup dialog box to return to the ODBC Data Source Administrator dialog box.

10. Click OK. The ODBC data source is configured.

You can use the MicroStrategy DB Query Tool to test whether data can be retrieved from the tables you created from the Excel file. For information on how to use the MicroStrategy DB Query Tool, see Using the DB Query Tool, page 155.

Create a database instance for an Excel file

To use an Excel file as a data source, you must create a database instance in MicroStrategy. This database instance is used to connect to the Excel file. Follow the steps below to create a database instance for an Excel file. It is recommended that you use Microsoft Excel 2000/2003 as the data source.
connection type for the database instance. For information on creating a database instance, see *Creating a database instance, page 197*.

Text files

A text file can be used as a data source in MicroStrategy. The information provided below explains how to prepare a text file for use with MicroStrategy and how to connect to the text file.

This data can be used as part of a MicroStrategy project in various ways. For example, you can integrate the text file data in your project using tools such as Architect, as described in the *Project Design Guide*. You can also use Freeform SQL and Query Builder to access your Excel data, as described in the *Advanced Reporting Guide*.

Prepare a text file as a valid data source

To use a text file as a data source, you must create and store the data in the text file so that it can be recognized in MicroStrategy as a set of tables that contain valid data.

**To prepare a text file as a valid data source**

This procedure assumes you have an existing text file that is formatted to be used as a data source through an ODBC connection.

1. Ensure that the text file is of type .txt or .csv.

2. Select a delimiter to separate the data in the text file into columns for a database table. The following procedure assumes a comma (,) is used as the delimiter character.

3. Ensure that the column names appear in the first row of the file and are delimited.

4. Save the text file on your machine. The file path is used as the data source directory in MicroStrategy.
Create a data source name for a text file

To use a text file as a data sources, you must create a DSN for the text file. This DSN is used in MicroStrategy to connect to the text file.

To create a DSN for a text file

1. From your machine’s Control Panel, select Administrative Tools, and then Data Sources (ODBC). The ODBC Data Source Administrator dialog box is displayed.

2. Select the System DSN tab and then click Add. The Create New Data Source dialog box is displayed.

3. Select MicroStrategy ODBC Driver for Text as your ODBC driver and then click Finish. The ODBC Text Driver Setup dialog box is displayed.

4. On the General tab, enter a Data Source Name (DSN).

5. In the Database Directory field, provide the file path of the directory where you saved the text file.

6. Select Comma as the Default Table Type. This assumes a comma (,) is used as the delimiter character for the text file.

7. Select the Column Names in First Line check box.

8. On the Advanced tab, click Define. The Define File dialog box is displayed.

9. Select the text file you want to define and click Open. The Define Table dialog box is displayed.

10. You must associate a table name with the text file. In the Table field within the Table Information area, type the name to be used for the table within MicroStrategy.

11. Select the Column Names in First Line check box.

12. Click Guess to display all the columns contained in this text file.

13. Click OK to return to the Define File dialog box.

14. Click Cancel to return to the ODBC Text Driver Setup dialog box.
15 Click **Apply** and then **OK** to return to the ODBC Data Source Administrator dialog box.

16 Click **OK**. Your data source for the text file is now configured.

You can use the MicroStrategy DB Query Tool to test whether data can be retrieved from the tables you created from the text file. For information on how to use the MicroStrategy DB Query Tool, see *Using the DB Query Tool, page 155.*

### Create a database instance for a text file

To use a text file as a data source, you must create a database instance in MicroStrategy. This database instance is used to connect to the text file. Follow the steps below to create a database instance for a text file. It is recommended that you use Generic DBMS as the data connection type for the database instance. For information on creating a database instance, see *Creating a database instance, page 197.*

### Configuring ODBC parameters with odbc.ini

The `odbc.ini` file is the configuration file that stores the definitions for all the ODBC DSNs in a UNIX or Linux environment. Therefore this section is not relevant to ODBC and DSN connections on Windows.

For information on what operating systems each ODBC driver is certified for, see *Certified ODBC drivers for MicroStrategy Intelligence Server, page 79.*

These ODBC DSNs are defined by specifying values for certain DSN parameters. This file is activated by the environment variable ODBCINI, and is required by all ODBC applications. By default, the `odbc.ini` file is installed in `HOME_PATH`, where `HOME_PATH` is the directory you specified as the home directory during installation on UNIX or Linux. It contains the definitions for the following MicroStrategy-branded ODBC drivers:

- MicroStrategy ODBC driver for Oracle Wire Protocol
- MicroStrategy ODBC Driver for DB2 Wire Protocol (DB2 UDB)
- MicroStrategy ODBC Driver for DB2 Wire Protocol (iSeries/DB2 for i)
- MicroStrategy ODBC Driver for DB2 z/OS
• MicroStrategy ODBC Driver for Greenplum Wire Protocol
• MicroStrategy ODBC Driver for Informix 8 (XPS)
• MicroStrategy ODBC Driver for Informix Wire Protocol
• MicroStrategy ODBC Driver for Sybase ASE Wire Protocol
• MicroStrategy ODBC Driver for Microsoft SQL Server
• MicroStrategy ODBC Driver for MySQL Wire Protocol
• MicroStrategy ODBC Driver for PostgreSQL Wire Protocol

This section describes commonly used DSN parameters for all MicroStrategy-branded ODBC drivers.

MicroStrategy supports ODBC drivers from other vendors that you can install separately. This involves manually defining the DSN parameters in the odbc.ini file.

This section also describes the DSN parameters for the following ODBC drivers from other vendors:

• ODBC Driver for Red Brick
• ODBC Driver for Sybase Adaptive Server IQ
• ODBC Driver for Teradata
• ODBC Driver for Netezza

Modification of the odbc.ini file is necessary to configure ODBC driver settings or full ODBC drivers that are not accessible through the Connectivity Wizard. However, caution should be taken when modifying the odbc.ini file as incorrect modifications can cause unintended functionality and errors. To see an example of a configuration using an odbc.ini file, refer to the odbc.ini.example file installed in HOME_PATH, where HOME_PATH is the directory you specified as the home directory during installation on UNIX or Linux.

You must use the correct syntax when modifying parameters for the odbc.ini file. Most of the parameters are of the form parameter name = parameter value, but some parameters require more complex syntax.
Refer to the table below for documentation standards related to parameter syntax.


<table>
<thead>
<tr>
<th>Syntax</th>
<th>Indicates</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{option1} \mid \text{option2} )</td>
<td>You must select one of the options. For example, if the syntax is ( {Y \mid N} ), you must type ( Y ), or ( N ). There can be two or more options, with each option separated by a pipe ((</td>
</tr>
<tr>
<td>[ ]</td>
<td>Anything inside square brackets ([]) is optional. For example, a list of servers can be represented syntactically as ( \text{Servers}=\text{Server1},\ldots ). The following would be valid syntaxes for the example above:  * ( \text{Servers}=\text{Server1} ) * ( \text{Servers}=\text{Server1},\text{Server2},\text{Server3} )</td>
</tr>
</tbody>
</table>

**Oracle Wire Protocol**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlternateServers</td>
<td>You can enable driver-side connection failover for the driver by specifying a list of available servers. When the primary database server is unavailable, connections to the available database servers are attempted sequentially. Syntax:  ( (\text{HostName}=\text{hostvalue}:\text{PortNumber}=\text{portvalue}:{\text{SID}=\text{sidvalue} \mid \text{ServiceName}=\text{servicevalue}}[,...]) )</td>
</tr>
<tr>
<td>ApplicationUsingThreads</td>
<td>You can specify whether the driver works with single-threaded or multi-threaded applications. You have the following options:  * 1: (Default) Use this setting to ensure the thread-safe driver works with multi-threaded applications.  * 0: Use this setting to prevent additional processing, which occurs due to ODBC thread-safety standards, for drivers that work with single-threaded applications.</td>
</tr>
<tr>
<td>AuthenticationMethod</td>
<td>Specifies which type of authentication the driver uses to authenticate a user. The selected type of authentication must be supported by the database server, otherwise the driver displays an error. You have the following options:  * 1: the driver sends the user ID in clear text and encrypts the password before sending it to the server to be authenticated.  * 3: the driver uses client authentication. The server does not provide any authentication; it relies on the client to authenticate the user.  * 4: the driver uses Kerberos authentication, which supports Windows Active Directory Kerberos and MIT Kerberos environments.  * 5: the driver uses Kerberos and user ID and password authentication. First, the user is authenticated using Kerberos and is then re-authenticated by the driver using the username and password.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CatalogOptions</td>
<td>You can specify whether result columns <code>REMARKS</code> and <code>COLUMN_DEF</code> return values or <code>SQL_NULL_DATA</code>. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) The <code>REMARKS</code> and <code>COLUMN_DEF</code> result columns return <code>SQL_NULL_DATA</code>.</td>
</tr>
<tr>
<td></td>
<td>• 1: The <code>REMARKS</code> and <code>COLUMN_DEF</code> result columns return their actual values.</td>
</tr>
<tr>
<td>ConnectionRetryCount</td>
<td>You can indicate the number of times this driver will attempt to connect to the primary or alternate servers (if enabled), if the initial attempt fails. The limits of this parameter are:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) No reconnection attempt is made.</td>
</tr>
<tr>
<td></td>
<td>• 65535: Maximum number of retries.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of a data source name.</td>
</tr>
<tr>
<td>Driver</td>
<td>Path where the ODBC driver being used is located.</td>
</tr>
<tr>
<td>EnableDescribeParam</td>
<td>You can enable the ODBC API function <code>SQLDescribeParam</code>. This function causes all parameters to be described with a <code>SQL_VARCHAR</code> data type. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) Disables the <code>SQLDescribeParam</code> function.</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default when using Microsoft Remote Data Objects to access data) Enables the <code>SQLDescribeParam</code> function.</td>
</tr>
<tr>
<td>EnableNcharSupport</td>
<td>You can enable support for the N-datatypes <code>NCHAR</code>, <code>NVARCHAR2</code>, and <code>NCLOB</code>. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) Disables support for <code>NCHAR</code>, <code>NVARCHAR2</code>, and <code>NCLOB</code>.</td>
</tr>
<tr>
<td></td>
<td>• 1: Enables support for <code>NCHAR</code>, <code>NVARCHAR2</code>, and <code>NCLOB</code>. These N-datatypes are described as <code>SQL_WCHAR</code>, <code>SQL_WVARCHAR</code>, and <code>SQL_WLONGVARCHAR</code>, respectively.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: This parameter is supported in Oracle 9i and later.</td>
</tr>
<tr>
<td>EnableStaticCursorsForLongData</td>
<td>You can enable support for columns with the long data type when using static cursors with your driver. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) Disables support for columns with the long data type when using static cursors.</td>
</tr>
<tr>
<td></td>
<td>• 1: Enables support for columns with the long data type when using static cursors. Use this option to persist a result set containing long data types into an XML file.</td>
</tr>
<tr>
<td>EncryptionMethod</td>
<td>Specifies how the driver encrypts data that is sent between the driver and the database server. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0 (Default): the data is not encrypted</td>
</tr>
<tr>
<td></td>
<td>• 1: the data is encrypted using SSL</td>
</tr>
<tr>
<td>GSSClient</td>
<td>The driver, by default, uses the GSS Client library packaged with the operating system to communicate with the Key Distribution Center. This parameter can be changed to use a different GSS Client library.</td>
</tr>
<tr>
<td>HostName</td>
<td>Name of the host machine where the database server resides. This may also be the IP address.</td>
</tr>
</tbody>
</table>
### HostNameInCertificate
Specifies whether or not the driver validates the Host Name in the certificate that is used in systems operating using SSL authentication. This parameter is ignored if the EncryptionMethod is set to 0, and if the value in ValidateServerCertificate is set to zero.

### IANAAppCodePage
You must provide a value for this parameter if your application is not enabled to support Unicode or you use a non-Unicode database character set. The value that you specify must be the value used for the code page used by your application. The driver on UNIX/Linux determines the value of the application’s code page by checking for an IANAAppCodePage value, in the following order:

1. In the connection string
2. In the DataSource section of the `odbc.ini` file
3. In the ODBC section of the `odbc.ini` file

The default value for this parameter provides support for most environments. If this parameter must be modified to support your code page, refer to the Data Direct documentation provided in the `Help\English\odbcHelp` folder. This folder location can be located in the MicroStrategy Common Files on Windows environments, and in the MicroStrategy install path on UNIX and Linux environments.

**Note:** The IANAAppCodePage connection string parameter replaces the AppCodePage connection string parameter in earlier versions of Connect for ODBC. The drivers are backward compatible with the AppCodePage parameter, but you must now use the IANAAppCodePage parameter. In this case, the parameter uses the value in the ODBC section of the `odbc.ini` file.

### KeyPassword
The password to access a particular key in the keystore.

### KeyStore
The path to the location of the keystore, which is the file that contains the list of the client certificates that are trusted by the server for Client Authentication with SSL.

### KeyStorePassword
The password to gain access to the keystore.

### LoadBalancing
You can enable client load balancing when connecting to primary and alternate servers. You have the following options:

- **0:** (Default) Disables client load balancing. Connection attempts to database servers are performed sequentially, starting with the primary server.
- **1:** Enables client load balancing. Connection attempts to database servers are performed in no particular order.

**Note:** Use the `AlternateServers` parameter to define alternate servers to connect to.

### LogonID
User ID used to log on to the database.

### Password
Password for the user account specified as the LogonID.

### PortNumber
Port number of the database instance.
### ProcedureRetResults
You can enable the driver to return result sets from stored procedure functions. You have the following options:
- 0: (Default) The driver does not return result sets from stored procedures.
- 1: The driver returns result sets from stored procedures.

### SID
Site Identifier of the Oracle instance you are trying to connect to.

### TNSNamesFile
You can specify the location of the TNSNAMES.ORA file. Make sure to type the entire path to the TNSNAMES.ORA file, including the file name itself. You can specify multiple TNSNAMES.ORA files.
You must set the Server Name parameter to use this option. Using the Server Name and TNSNamesFiles parameters disables the HostName, PortNumber, SID, and Service Name fields.
Syntax:
\[
(filelocation1[,\ldots])
\]

### TrustStore
The path for the location to the truststore, which is the file that contains the list of certificate authorities that are trusted by the machine for SSL server authentication.

### TrustStorePassword
The password to gain access to the truststore.

### ValidateServerCertificate
Specifies whether or not the driver validates the server’s security certificate during SSL authentication. You have the following options:
- 0: the certificate is not validated
- 1 (Default): the certificate is validated

### Parameter | Description
--- | ---
ProcedureRetResults | You can enable the driver to return result sets from stored procedure functions. You have the following options:
- 0: (Default) The driver does not return result sets from stored procedures.
- 1: The driver returns result sets from stored procedures.

### SID | Site Identifier of the Oracle instance you are trying to connect to.

### TNSNamesFile | You can specify the location of the TNSNAMES.ORA file. Make sure to type the entire path to the TNSNAMES.ORA file, including the file name itself. You can specify multiple TNSNAMES.ORA files.
You must set the Server Name parameter to use this option. Using the Server Name and TNSNamesFiles parameters disables the HostName, PortNumber, SID, and Service Name fields.
Syntax:
\[
(filelocation1[,\ldots])
\]

### TrustStore | The path for the location to the truststore, which is the file that contains the list of certificate authorities that are trusted by the machine for SSL server authentication.

### TrustStorePassword | The password to gain access to the truststore.

### ValidateServerCertificate | Specifies whether or not the driver validates the server’s security certificate during SSL authentication. You have the following options:
- 0: the certificate is not validated
- 1 (Default): the certificate is validated

### Parameter | Description
--- | ---
AuthenticationMethod | Specifies which type of authentication the driver uses to authenticate a user. The selected type of authentication must be supported by the database server, otherwise the driver displays an error.
You have the following options:
- 0 (Default): the driver sends the user ID and password to the server in clear text to be authenticated.
- 1: the driver sends the user ID in clear text and encrypts the password before sending it to the server to be authenticated.
- 2: the driver encrypts both the user ID and the password before sending it to the server to be authenticated.
- 3: the driver uses client authentication. The server does not provide any authentication; it relies on the client to authenticate the user.
- 4: the driver users Kerberos authentication, which supports Windows Active Directory Kerberos and MIT Kerberos environments.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EncryptionMethod</td>
<td>Specifies the way the driver encrypts data that is sent between the driver and the database server. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0 (Default): the data is not encrypted</td>
</tr>
<tr>
<td></td>
<td>• 1: the data is encrypted using SSL (supported only on DB2 for iSeries/DB2 for i)</td>
</tr>
<tr>
<td></td>
<td>• 2: the data is encrypted using the DB2 encryption protocol (supported only on DB2 for Linux/UNIX/Windows and DB2 for z/OS)</td>
</tr>
<tr>
<td></td>
<td>The AuthenticationMethod must be set to 0, 1, or 2 to use DB2 encryption. If the Database server does not support the type of encryption, it displays an error and the connection fails.</td>
</tr>
<tr>
<td>GSS Client</td>
<td>The driver, by default, uses the GSS Client library that comes with the operating system to communicate with the Key Distribution Center. This parameter can be changed to use a different GSS Client library.</td>
</tr>
<tr>
<td>HostNameInCertificate</td>
<td>Specifies whether or not the driver validates the Host Name in the certificate that is used in systems operating using SSL authentication. This parameter is ignored if the EncryptionMethod is not set to 1, and if the value in ValidateServerCertificate is set to zero.</td>
</tr>
<tr>
<td>TrustStore</td>
<td>The path for the location to the truststore, which is the file that contains the list of certificate authorities that are trusted by the machine for SSL server authentication.</td>
</tr>
<tr>
<td>TrustStorePassword</td>
<td>Password to gain access to the truststore.</td>
</tr>
<tr>
<td>ValidateServerCertificate</td>
<td>Specifies whether or not the driver validates the server’s security certificate during SSL authentication. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: the certificate is not validated</td>
</tr>
<tr>
<td></td>
<td>• 1 (Default): the certificate is validated</td>
</tr>
<tr>
<td>XMLDescribeType</td>
<td>Specifies what SQL Data Type the SQLGetTypeInfo() returns for the XML Data type. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• -4: The description SQL_LONGVARBINARY is used for DB2 XML data type columns.</td>
</tr>
<tr>
<td></td>
<td>• -10 (Default): The description SQL_WLONGVARCHAR is used for DB2 XML data type columns.</td>
</tr>
<tr>
<td>Driver</td>
<td>Path where the ODBC driver being used is located.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of this data source.</td>
</tr>
<tr>
<td>LogonID</td>
<td>Default user ID to log on to the database. On UNIX/Linux, the LogonID value is your UNIX/Linux user ID.</td>
</tr>
<tr>
<td>Password</td>
<td>Password for the user account specified as the LogonID.</td>
</tr>
<tr>
<td>AlternateID</td>
<td>Default qualifier used at connection time for any unqualified object names used in SQL statements. You must set the DB2 permissions to SYSADM. DB2 UDB V5R1 on iSeries does not support this parameter.</td>
</tr>
<tr>
<td>IpAddress</td>
<td>Specifies the IP address of the machine where the catalog tables are stored, as either a numeric address or address name. If you enter an address name, the address name is retrieved from the workstation’s HOSTS file or in a DNS server.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Database</td>
<td>Name of the database to which to connect.</td>
</tr>
<tr>
<td>Collection</td>
<td>Name that identifies a logical group of database objects. This parameter is valid only if you are connecting to DB2 for z/OS (formerly OS/390) or iSeries/DB2 for i. On iSeries/DB2 for i, it is also the current Schema. These objects include the DataDirect Connect DB2 Wire Protocol driver for ODBC packages.</td>
</tr>
<tr>
<td>Location</td>
<td>Specifies the DB2 location name. Use the name defined on the DB2 Server. This parameter is valid only if you are connecting to DB2 for z/OS (formerly OS/390) or iSeries/DB2 for i.</td>
</tr>
<tr>
<td>TcpPort</td>
<td>Specifies the port number used by the DB2 database instance on the server host machine.</td>
</tr>
</tbody>
</table>
| GrantExecute| You can grant execute privileges on the package you specify as the GrantAuthID. You have the following options:  
  • 1: (Default) Privileges are granted.  
  • 0: Privileges are not granted.                                                                                                   |
| GrantAuthid | You can indicate to whom execute privileges are granted. By default, this parameter is set to PUBLIC.  
  **Note:** This parameter is ignored if GrantExecute=0.                                                                                         |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DefaultIsolationLevel</td>
<td>Method by which locks are acquired and released by the system. ODBC isolation levels map to DB2 isolation levels as follows:</td>
</tr>
<tr>
<td></td>
<td><strong>ODBC</strong></td>
</tr>
<tr>
<td></td>
<td><strong>DB2</strong></td>
</tr>
<tr>
<td>Read Uncommitted</td>
<td>Uncommitted Read</td>
</tr>
<tr>
<td>Read Committed</td>
<td>Cursor Stability</td>
</tr>
<tr>
<td>Repeatable Read</td>
<td>Read Stability</td>
</tr>
<tr>
<td>Serializable</td>
<td>Repeatable Read</td>
</tr>
<tr>
<td>Valid values are:</td>
<td>• 0 - READ_UNCOMMITTED: Other processes can read from the database. Modified data is locked until the end of the transaction.</td>
</tr>
<tr>
<td></td>
<td>• 1 - READ_COMMITTED: (Default) Other processes cannot modify a row if the cursor is on the row or modify records that your application has modified, until your application commits or terminates. Your application cannot read a modified record that has not been committed by another process.</td>
</tr>
<tr>
<td></td>
<td>• 2 - REPEATABLE_READ: Other process cannot access data that your application has read or modified, until the end of the transaction.</td>
</tr>
<tr>
<td></td>
<td>• 3 - SERIALIZABLE: Other processes cannot modify records, including phantom records, that are read or modified by your application until your program commits or terminates. Your application is prevented from reading modified records that have not been committed by another process. If your application opens the same query during a single unit of work, the results table is identical to the previous table. However, results table can contain updates made by your application.</td>
</tr>
<tr>
<td></td>
<td>• 4 - NONE: Your program can read modified records even if they have not been committed by another process. This level can only be set in the data source, not from the application.</td>
</tr>
<tr>
<td>Note:</td>
<td>You must use this level if you have a collection on iSeries/DB2 for i with journaling enabled.</td>
</tr>
<tr>
<td>DynamicSections</td>
<td>Specifies how many statements the DB2 Wire Protocol driver package can prepare for each user. The default is 200.</td>
</tr>
<tr>
<td>CatalogSchema</td>
<td>Specifies the name of a valid DB2 schema to use for Catalog functions. If you do not specify a value, the parameter defaults to the following values:</td>
</tr>
<tr>
<td></td>
<td>• SYSIBM: Default for z/OS.</td>
</tr>
<tr>
<td></td>
<td>• QSYS2: Default for iSeries/DB2 for i.</td>
</tr>
<tr>
<td></td>
<td>• SYSCAT: Default when connected to UNIX/Linux.</td>
</tr>
<tr>
<td>PackageCollection</td>
<td>Specifies the collection or location name where the driver creates and searches for the bind packages. The default is NULLID.</td>
</tr>
<tr>
<td>PackageOwner</td>
<td>Specifies the DB2 user account assigned to the package. This user must have authority to run all the SQL statements in the package.</td>
</tr>
</tbody>
</table>
### Informix 8 (XPS)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>UseCurrentSchema</strong></td>
<td>You can restrict results to the tables in the current schema when a catalog function call is made either without specifying a schema or when specifying the schema as the wildcard character %. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) Results can include tables outside of the current schema.</td>
</tr>
<tr>
<td></td>
<td>• 1: Results are restricted to tables in the current schema. Restricting results to tables in the current schema improves the performance of calls that do not specify a schema.</td>
</tr>
<tr>
<td><strong>ApplicationUsingThreads</strong></td>
<td>You can specify whether the driver works with single-threaded or multi-threaded applications. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default) Use this setting to ensure the thread-safe driver works with multi-threaded applications.</td>
</tr>
<tr>
<td></td>
<td>• 0: Use this setting to prevent additional processing, which occurs due to ODBC thread-safety standards, for drivers that work with single-threaded applications.</td>
</tr>
<tr>
<td><strong>WithHold</strong></td>
<td>Specifies the cursor behavior for the application used with this data source. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default) DB2 leaves all cursors open after a commit or rollback. (SQLGetInfo( ) returns SQL_CB_PRESERVE for SQL_COMMIT_CURSOR_BEHAVIOR.)</td>
</tr>
<tr>
<td></td>
<td>• 0: DB2 closes all open cursors after a commit or rollback. (SQLGetInfo( ) returns SQL_CB_DELETE.)</td>
</tr>
</tbody>
</table>

### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ApplicationUsingThreads</td>
<td>You can specify whether the driver works with single-threaded or multi-threaded applications. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default) Use this setting to ensure the thread-safe driver works with multi-threaded applications.</td>
</tr>
<tr>
<td></td>
<td>• 0: Use this setting to prevent additional processing, which occurs due to ODBC thread-safety standards, for drivers that work with single-threaded applications.</td>
</tr>
<tr>
<td>CancelDetectInterval</td>
<td>Specifies how often (in seconds) the driver checks whether a query has been canceled. When the driver detects that SQLCancel has been issued by the MicroStrategy Platform, the query is canceled. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) Queries are not canceled even if SQLCancel is issued.</td>
</tr>
<tr>
<td></td>
<td>• {5</td>
</tr>
</tbody>
</table>
## Parameter Description

<table>
<thead>
<tr>
<th><strong>Parameter</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CursorBehavior</strong></td>
<td>Determines the behavior of cursors at the end of each transaction. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) Cursors are closed at the end of each transaction.</td>
</tr>
<tr>
<td></td>
<td>• 1: Cursors are preserved, meaning they are held at the current position at the end of each transaction. Preserving cursor position can slow down your database operations.</td>
</tr>
<tr>
<td><strong>Database</strong></td>
<td>Name of the database to which to connect.</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>Description of this data source.</td>
</tr>
<tr>
<td><strong>Driver</strong></td>
<td>Path where the ODBC driver being used is located.</td>
</tr>
<tr>
<td><strong>EnableInsertCursors</strong></td>
<td>You can enable Insert cursors. You have the following options.</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default) Insert cursors are enabled. Insert cursors improve performance during multiple Insert operations using the same statement. Insert data can be buffered in memory before being written to disk.</td>
</tr>
<tr>
<td></td>
<td>• 0: Insert cursors are disabled.</td>
</tr>
<tr>
<td><strong>GetDBListFromInformix</strong></td>
<td>Specifies whether the database list is returned from the Informix server or from the database list the user entered during driver setup.</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default) The database list is returned from the Informix server.</td>
</tr>
<tr>
<td></td>
<td>• 0: The database list is returned from the database list the user entered during driver setup.</td>
</tr>
<tr>
<td><strong>HostName</strong></td>
<td>Name of the machine on which the Informix server resides.</td>
</tr>
<tr>
<td><strong>LogonID</strong></td>
<td>Logon ID used to connect to your database.</td>
</tr>
<tr>
<td><strong>Password</strong></td>
<td>Password used to connect to your database.</td>
</tr>
<tr>
<td><strong>Protocol</strong></td>
<td>Specifies the protocol used to communicate with the server. You can specify one or more values; separate the names with commas. You can enter any of the following protocols:</td>
</tr>
<tr>
<td></td>
<td>• olsocspx</td>
</tr>
<tr>
<td></td>
<td>• olsoctcp</td>
</tr>
<tr>
<td></td>
<td>• onsocspx</td>
</tr>
<tr>
<td></td>
<td>• onsoctcp</td>
</tr>
<tr>
<td></td>
<td>• seipcpip</td>
</tr>
<tr>
<td></td>
<td>• sesocspx</td>
</tr>
<tr>
<td></td>
<td>• sesoctcp</td>
</tr>
<tr>
<td></td>
<td>• onliltcp</td>
</tr>
<tr>
<td><strong>ServerName</strong></td>
<td>Name of the server running the Informix instance.</td>
</tr>
<tr>
<td><strong>Service</strong></td>
<td>Name of the service being used by Informix on the host machine.</td>
</tr>
<tr>
<td><strong>TrimBlankFromIndexName</strong></td>
<td>You can specify whether to trim leading space from a system-generated index name. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default) The driver trims the leading space.</td>
</tr>
<tr>
<td></td>
<td>• 0: The driver does not trim the space.</td>
</tr>
</tbody>
</table>
# Informix Wire Protocol

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ApplicationUsingThreads</td>
<td>You can specify whether the driver works with single-threaded or multi-threaded applications. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default) Use this setting to ensure the thread-safe driver works with multi-threaded applications.</td>
</tr>
<tr>
<td></td>
<td>• 0: Use this setting to prevent additional processing, which occurs due to ODBC thread-safety standards, for drivers that work with single-threaded applications.</td>
</tr>
<tr>
<td>CancelDetectInterval</td>
<td>Specifies how often (in seconds) the driver checks whether a query has been canceled. When the driver detects that SQLCancel has been issued by the MicroStrategy Platform, the query is canceled. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) Queries are not canceled even if SQLCancel is issued. n[5</td>
</tr>
<tr>
<td>Database</td>
<td>Name of the database to which to connect.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the data source.</td>
</tr>
<tr>
<td>Driver</td>
<td>Path where the ODBC driver being used is located.</td>
</tr>
<tr>
<td>HostName</td>
<td>Name or IP address of the machine on which the Informix server resides.</td>
</tr>
<tr>
<td>LogonID</td>
<td>User ID for connecting to the database.</td>
</tr>
<tr>
<td>Password</td>
<td>Password for connecting to your database.</td>
</tr>
<tr>
<td>PortNumber</td>
<td>Port number of the server listener.</td>
</tr>
<tr>
<td>ServerName</td>
<td>Name of the server running the Informix instance.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: This can be different from the HostName. Verify what your ServerName is with the Informix database administrator.</td>
</tr>
<tr>
<td>TrimBlankFromIndexName</td>
<td>You can specify whether to trim leading space from a system-generated index name. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default) The driver trims leading space.</td>
</tr>
<tr>
<td></td>
<td>• 0: The driver does not trim leading space.</td>
</tr>
<tr>
<td>UseDelimitedIdentifiers</td>
<td>Specifies how the server interprets double quotation marks(&quot;&quot;) when used in SQL statements. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0 (Default): strings within double quotation marks are interpreted as string literals.</td>
</tr>
<tr>
<td></td>
<td>• 1: strings within double quotation marks are interpreted as identifiers.</td>
</tr>
</tbody>
</table>
## Sybase ASE Wire Protocol

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| ApplicationUsingThreads   | You can specify whether the driver works with single-threaded or multi-threaded applications. You have the following options:  
  - 1: (Default) Use this setting to ensure the thread-safe driver works with multi-threaded applications.  
  - 0: Use this setting to prevent additional processing, which occurs due to ODBC thread-safety standards, for drivers that work with single-threaded applications. |
| Authentication Method     | Specifies which type of authentication the driver uses to authenticate a user. The selected type of authentication must be supported by the database server, otherwise the driver displays an error.  
  You have the following options:  
  - 0 (Default): the driver sends the user ID and password to the server in clear text to be authenticated.  
  - 1: the driver sends the user ID in clear text and encrypts the password before sending it to the server to be authenticated.  
  - 4: the driver uses Kerberos authentication, which supports Windows Active Directory Kerberos and MIT Kerberos environments. |
| Charset                   | Specifies the name of a character set installed on the Sybase server. The default is the setting on the Sybase server. For Sybase ASE 12.5 and higher, set this parameter to UTF-8 to support Unicode SQL types. Refer to Sybase documentation for a list of valid character sets. |
| Database                  | Name of the database to use.                                                                                                                                                                                |
| Description               | Description of the data source.                                                                                                                                                                              |
| Driver                    | Path where the ODBC driver being used is located.                                                                                                                                                           |
| EnableDescribeParam       | You can enable the ODBC API function SQLDescribeParam. This function causes all parameters to be described with a SQL_VARCHAR data type. You have the following options:  
  - 0: (Default) Disables the SQLDescribeParam function.  
  - 1: (Default when using Microsoft Remote Data Objects to access data) Enables the SQLDescribeParam function. |
| EnableQuotedIdentifiers   | You can enable quoted identifiers. You have the following options:  
  - 0: (Default) Disables quoted identifiers. Applications that generate statements with quoted identifiers encounter errors.  
  - 1: Enables quoted identifiers. Double quotes are only allowed around identifiers in SQL statements such as column and table names. For example, an ID column for the Category attribute appears as "CATEGORY_ID" in the SQL statement. |
| EncryptionMethod          | Specifies the way the driver encrypts data that is sent between the driver and the database server. You have the following options:  
  - 0 (Default): the data is not encrypted  
  - 1: the data is encrypted using SSL |
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSSClient</td>
<td>The driver, by default, uses the GSS Client library that comes with the operating system to communicate with the Key Distribution Center. This parameter can be changed to use a different GSS Client library.</td>
</tr>
<tr>
<td>HostNameInCertificate</td>
<td>Specifies whether or not the driver validates the Host Name in the certificate that is used in systems operating using SSL authentication. This parameter is ignored if the EncryptionMethod is not set to SSL, and if the value in ValidateServerCertificate is set to zero.</td>
</tr>
<tr>
<td>Language</td>
<td>Specifies the national language installed on the Sybase server. The default is English.</td>
</tr>
<tr>
<td>LogonID</td>
<td>User ID for connecting to the database.</td>
</tr>
<tr>
<td>NetworkAddress</td>
<td>Specifies the network address. The format depends on the network protocol chosen under Network Library Name and the Sybase server. If you are choosing Windows Sockets (Winsock), then the format is: {IP address</td>
</tr>
<tr>
<td>OptimizePrepare</td>
<td>Specifies how stored procedures are prepared when using SQLPrepare. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: Stored procedures are created for every call to SQLPrepare. Decreased performance occurs when statements that do not contain parameters are processed.</td>
</tr>
<tr>
<td></td>
<td>• 1: (Default) Stored procedures are created only if the statement contains parameters. Otherwise, the statement for the stored procedure is cached and run when SQLEXecute is called.</td>
</tr>
<tr>
<td></td>
<td>• 2: Stored procedures are never created. The statement for the stored procedure is cached and run when SQLEXecute is called. Any syntax errors or other errors are returned when SQLEXecute is called.</td>
</tr>
<tr>
<td></td>
<td>• 3: Stored procedures are never created. This option is identical to option 2 except that any syntax errors or other errors are returned when SQLPrepare is called instead of SQLEXecute.</td>
</tr>
<tr>
<td>Password</td>
<td>Password for connecting to your database.</td>
</tr>
<tr>
<td>RaiseErrorPositionBehav ior</td>
<td>You can configure when errors are returned and where the cursor is positioned after encountering a raiseerror. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) Raiseerror is handled separately from surrounding statements. Errors are returned when raiseerror is processed via SQLEXecute, SQLEXecDirect, or SQLMoreResults.</td>
</tr>
<tr>
<td></td>
<td>• 1: (Microsoft-compatible) RaiseError is returned when the next statement is processed and the cursor is positioned on the first row of the subsequent result set. Multiple RaiseErrors may be returned on a single execute.</td>
</tr>
<tr>
<td>SelectMethod</td>
<td>You can enable database cursors for SELECT statements. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: (Default) Database cursors are enabled. Creating database cursors requires additional overhead, which can negatively affect performance when executing a large number of sequential SELECT statements.</td>
</tr>
<tr>
<td></td>
<td>• 1: Database cursors are disabled for SELECT statements, thus limiting the database to one active statement at a time.</td>
</tr>
</tbody>
</table>
### Configuring ODBC parameters with odbc.ini

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServicePrincipalName</td>
<td>The Sybase server principal name used for Kerberos authentication. If no name is specified, then the driver defaults to the value for NetworkAddress. This parameter is ignored if an authentication type other than Kerberos is used.</td>
</tr>
<tr>
<td>TrustStore</td>
<td>The path for the location to the truststore, which is the file that contains the list of certificate authorities that are trusted by the machine for SSL server authentication.</td>
</tr>
<tr>
<td>TrustStorePassword</td>
<td>The password to gain access to the truststore.</td>
</tr>
<tr>
<td>ValidateServerCertificate</td>
<td>Specifies whether or not the driver validates the server’s security certificate during SSL authentication. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• 0: the certificate is not validated</td>
</tr>
<tr>
<td></td>
<td>• 1(Default): the certificate is validated</td>
</tr>
</tbody>
</table>

### SQL Server Wire Protocol

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>Network address of the SQL Server. The address should be the network name of the server, but it can be a TCP/IP port and socket address. For example, on TCP/IP it can be 199.199.199.5; 1433; or MYSVR, 1433.</td>
</tr>
<tr>
<td>AnsiNPW</td>
<td>You can enable ANSI-defined behaviors.</td>
</tr>
<tr>
<td></td>
<td>• Yes: ANSI-defined behaviors are enabled to handle NULL comparisons, character data padding, warnings, and NULL concatenation. Modify this option to No if trailing blank spaces appear to be truncated.</td>
</tr>
<tr>
<td></td>
<td>• No: ANSI-defined behaviors are disabled.</td>
</tr>
<tr>
<td>Database</td>
<td>Database to use.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the data source.</td>
</tr>
<tr>
<td>Driver</td>
<td>Path where the ODBC driver being used is located.</td>
</tr>
<tr>
<td>LogonID</td>
<td>User ID for connecting to the database.</td>
</tr>
<tr>
<td>Password</td>
<td>Password for connecting to your database.</td>
</tr>
<tr>
<td>QuotedId</td>
<td>You can choose to handle QUOTED_IDENTIFIERS as defined by SQL-92 or Transact_SQL rules. You have the following options:</td>
</tr>
<tr>
<td></td>
<td>• Yes: QUOTED_IDENTIFIERS are enabled. SQL Server uses SQL-92 rules for quotation marks in SQL statements.</td>
</tr>
<tr>
<td></td>
<td>• No: QUOTED_IDENTIFIERS are disabled. SQL Server uses the legacy Transact_SQL rules for quotation marks in SQL statements.</td>
</tr>
</tbody>
</table>
If you are using Microsoft SQL server 2005 and it is configured for snapshot isolation, you can allow the application to use the snapshot isolation level. You have the following choices:

- 0 (Default): the application uses the serializable isolation level when the isolation level is set to serializable.
- 1: The application uses the snapshot isolation level when the isolation level is set to serializable.

---

### MySQL Wire Protocol

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlternateServers</td>
<td>You can enable driver-side connection failover for the driver by specifying a list of available servers. When the primary database server is unavailable, connections to the available database servers are attempted sequentially. Syntax: (HostName=hostvalue:PortNumber=portvalue: [SID=sidvalue</td>
</tr>
</tbody>
</table>
| ApplicationUsingThreads | You can specify whether the driver works with single-threaded or multi-threaded applications. You have the following options:  
  • 1: (Default) Use this setting to ensure the thread-safe driver works with multi-threaded applications.  
  • 0: Use this setting to prevent additional processing, which occurs due to ODBC thread-safety standards, for drivers that work with single-threaded applications. |
| ConnectionRetryCount | You can indicate the number of times this driver will attempt to connect to the primary or alternate servers (if enabled), if the initial attempt fails. The limits of this parameter are:  
  • 0: (Default) No attempt is made to reconnect.  
  • 65535: Maximum number of retries. |
| ConnectionRetryDelay | You can specify the amount of time, in seconds, that the driver delays between attempts to connect to the primary server and alternate servers, if specified. Integers within the following limits are valid:  
  • 0: There is no delay between attempts to reconnect.  
  • 3 (default): The default delay is three seconds.  
  • 65535: The maximum delay between attempts to reconnect.  
  This parameter only has an effect if the ConnectionRetryCount parameter is set to a value other than 0. |
| Database         | Specifies the database to use. |
### Connecting to Databases: ODBC and DSNs

#### Configuring ODBC parameters with odbc.ini

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DefaultLongDataBuffLen</td>
<td>You can specify the maximum data length, in bytes, that is fetched from Long/LOB columns. The value must be in multiples of 1024, and is multiplied by 1024 to define the maximum data length in bytes. For example, the default for this parameter is 1024, which is multiplied by 1024, for a total of 1048576 bytes, or 1 megabyte. This parameter also resizes any data passed in a Long/LOB SQL_DATA_AT_EXEC parameter to fit the data length specified, and chunks any data exceeding this data length limit.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the data source.</td>
</tr>
<tr>
<td>Driver</td>
<td>Path where the ODBC driver being used is located.</td>
</tr>
</tbody>
</table>
| EnableDescribeParam           | You can enable the ODBC API function SQLDescribeParam. This function causes all parameters to be described with a SQL VARCHAR data type. You have the following options:  
  • 0: (Default) Disables the SQLDescribeParam function.  
  • 1: (Default when using Microsoft Remote Data Objects to access data) Enables the SQLDescribeParam function. |
| HostName                      | Name of the host machine where the database server resides. This may also be the IP address.                                                                                                                  |
| InteractiveClient             | Determines which variable is used to determine the amount of time that a connection can idle until the server disconnects it. The possible values are:  
  • 0 (Default): the driver uses the value for the global wait_timeout variable to initialize the wait_timeout session  
  • 1: the driver uses the global interactive_timeout variable value to initialize the wait_time session  
  The wait_timeout variable, controlled by InteractiveClient, can be modified after the connection has been made. |
| LoadBalancing                 | You can enable client load balancing when connecting to primary and alternate servers. You have the following options:  
  • 0: (Default) disables client load balancing. Connection attempts to database servers are performed sequentially, starting with the primary server.  
  • 1: enables client load balancing. Connection attempts to database servers are performed in no particular order.  
  Note: Use the AlternateServers parameter to define alternate servers to connect to. |
| LogonID                       | User ID used to log on to the database.                                                                                                                                                                     |
| Password                      | Password for connecting to your database.                                                                                                                                                                   |
| PortNumber                    | Port number of the server listener. Default is 3306.                                                                                                                                                       |
Configuring ODBC parameters with odbc.ini
© 2011 MicroStrategy, Inc.

PostgreSQL and Greenplum Wire Protocols

The ODBC drivers for PostgreSQL and Greenplum Wire Protocols can be configured using many of the same parameters, including the parameters listed below:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| AlternateServers   | You can enable driver-side connection failover for the driver by specifying a list of available servers. When the primary database server is unavailable, connections to the available database servers are attempted sequentially. Syntax:  
(HostName=hostvalue:PortNumber=portvalue:  
{SID=sidvalue |ServiceName=servicevalue}[,,...]) |
| ApplicationUsingThreads | You can specify whether the driver works with single-threaded or multi-threaded applications. You have the following options:  
• 1: (Default) Use this setting to ensure the thread-safe driver works with multi-threaded applications.  
• 0: Use this setting to prevent additional processing, which occurs due to ODBC thread-safety standards, for drivers that work with single-threaded applications. |
### Parameter | Description
--- | ---
ConnectionRetryCount | You can indicate the number of times this driver will attempt to connect to the primary or alternate servers (if enabled), if the initial attempt fails. The limits of this parameter are:
- 0: (Default) No attempt is made to reconnect.
- 65535: Maximum number of retries.

ConnectionRetryDelay | You can specify the amount of time, in seconds, that the driver delays between attempts to connect to the primary server and alternate servers, if specified. Integers within the following limits are valid:
- 0: There is no delay between attempts to reconnect.
- 3 (default): The default delay is three seconds.
- 65535: The maximum delay between attempts to reconnect.
This parameter only has an effect if the ConnectionRetryCount parameter is set to a value other than 0.

Database | Specifies the database to use.
Description | Description of the data source.
Driver | Path where the ODBC driver being used is located.
EnableDescribeParam | You can enable the ODBC API function SQLDescribeParam. You have the following options:
- 0: (Default) Disables the SQLDescribeParam function.
- 1: (Default when using Microsoft Remote Data Objects to access data) Enables the SQLDescribeParam function.

HostName | Name of the host machine where the database server resides. This may also be the IP address.
LoadBalancing | You can enable client load balancing when connecting to primary and alternate servers. You have the following options:
- 0: (Default) disables client load balancing. Connection attempts to database servers are performed sequentially, starting with the primary server.
- 1: enables client load balancing. Connection attempts to database servers are performed in no particular order.
**Note:** Use the AlternateServers parameter to define alternate servers to connect to.

LogonID | User ID used to log on to the database.
Password | Password for connecting to your database.
PortNumber | Port number of the server listener. Default is 5432.
### ODBC Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| IANAAppCodePage            | You must provide a value for this parameter if your application is not enabled to support Unicode or you use a non-Unicode database character set. The value you specify must be the value used for the code page used by your application. The driver on UNIX/Linux determines the value of the application’s code page by checking for an IANAAppCodePage value, in the following order:  
1 In the connection string  
2 In the DataSource section of the odbc.ini file  
3 In the ODBC section of the odbc.ini file  
The default value for this parameter provides support for most environments. If this parameter must be modified to support your code page, refer to the Data Direct documentation provided in the Help\English\odbcHelp folder. This folder location can be located in the MicroStrategy Common Files on Windows environments, and in the MicroStrategy install path on UNIX and Linux environments.  
**Note:** The IANAAppCodePage connection string parameter replaces the AppCodePage connection string parameter in earlier versions of Connect for ODBC. The drivers are backward compatible with the AppCodePage parameter, but you must now use the IANAAppCodePage parameter. In this case, the parameter uses the value in the ODBC section of the odbc.ini file. |
| InstallDir                 | Location where ODBC drivers are installed.                                                                                                  |
## Red Brick 6.x

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATABASE</td>
<td>Name of the database to which to connect.</td>
</tr>
<tr>
<td>Driver</td>
<td>Path where the ODBC driver being used is located.</td>
</tr>
<tr>
<td>PWD</td>
<td>Password used to connect to your database.</td>
</tr>
<tr>
<td>RB_CONFIG</td>
<td>Location where the ODBC driver is installed.</td>
</tr>
<tr>
<td>SERVER</td>
<td>Host name and port number of the server that runs the database.</td>
</tr>
<tr>
<td>UID</td>
<td>User ID to log on to the database.</td>
</tr>
</tbody>
</table>

## Sybase IQ

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoStop</td>
<td>Specifies whether the database should continue running after connections are disconnected.</td>
</tr>
<tr>
<td>CommLinks</td>
<td>Host name and port number of the server that runs the database. The value should be as follows: tcpip(host=X.X.X.X;port=x)</td>
</tr>
<tr>
<td>DatabaseName</td>
<td>Name of the database to connect.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the data source.</td>
</tr>
<tr>
<td>Driver</td>
<td>Path where the ODBC driver being used is located.</td>
</tr>
<tr>
<td>EngineName</td>
<td>Name of the database Server to connect.</td>
</tr>
<tr>
<td>PWD</td>
<td>Password used to connect to your database.</td>
</tr>
<tr>
<td>UID</td>
<td>User ID to connect to the database.</td>
</tr>
</tbody>
</table>
### Teradata Server

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| CharSet           | Specifies the default character set for the session. The default character set for Single Byte Character Sets (SBCS) is ASCII. For DBCS character sets, the value in the KanjiFormat option should match the character set name specified for this option. You have the following options:  
  • blank (default for KanjiFormat)  
  • ASCII (for SBCS)  
  • UTF8  
  • SCHEBCDIC935_2I, TCHEBCKIC937_3IB  
  • TCHBIG5_1RO, SDTCHEBIG5_3RO  
  • SCHEGB2312_1TO, SDSCHGB2312_2TO  
  • SDSCHEBCDIC935_6I, SDTCHEBCDIC937_7IB  
  • SDKANJIEBCDIC5026_4I, SDKANJIEBCDIC5035_4I  
  • SDKANJIEUC_1U3, SDKANJISJIS_1S3  
  • HANGULEBCKIC933_1II, HANGULKSC5601_2R4  
  • SDHANGULEBCKIC933_5II, SDHANGULKSC5601_4R4  
  • KANJISJIS_OS, KANJIEUC_OU  |
| Database          | Name of the database to connect.                                                                                                                  |
| DateTimeFormat    | Specifies whether the ODBC driver should handle DATEs, TIMEs, and TIMESTAMPs as integer values or ANSI-compatible strings.  
  • III specifies that all three should be handled as integer values.  
  • AAA specifies that they should be handled as ANSI-compatible strings. |
| DBCName           | Identifies the host by name (alias) or IP Address.                                                                                             |
| DefaultDatabase   | Name of the default database.                                                                                                                    |
| Description       | Description of the data source.                                                                                                                   |
| Driver            | Path where the ODBC driver being used is located.                                                                                              |
| EnableExtendedStmtInfo | Specifies whether to enable extended statement information for databases that support extended statement information, which includes Teradata V2R6.2 and later versions. Enabling extended statement information also supports parameterized queries. You have the following options:  
  • Yes: Enables extended statement information, which includes supporting the SQLDescribeParam API function. Enabling extended statement information also supports parameterized queries.  
  • No: Disables extended statement information and parameterized queries. |
IANAAppCodePage | Used with the transliteration system. The code page that you specify must be the same as the code page used by your application. The driver on UNIX/Linux determines the value of the application's code page by checking for an IANAAppCodePage value, in the following order:
1. In the connection string
2. In the DataSource section of the system file (odbc.ini)
3. In the ODBC section of the system file (odbc.ini)
The default value for this parameter provides support for most environments. If this parameter must be modified to support your code page, refer to the Data Direct documentation provided in the Help\English\odbcHelp folder. This folder location can be located in the MicroStrategy Common Files on Windows environments, and in the MicroStrategy install path on UNIX and Linux environments.

**Note**: The IANAAppCodePage connection string parameter replaces the AppCodePage connection string parameter in earlier versions of Connect for ODBC. The drivers are backward compatible with the AppCodePage parameter, but you must now use the IANAAppCodePage parameter. In this case, the parameter uses the value in the ODBC section of the odbc.ini file.

LastUser | Remembers the connection for the last user.

NoScan | You can enable parsing of SQL statements in the driver. You have the following options:
- **Yes**: (Default) Parsing is disabled. The parser in the driver is bypassed and the SQL statement is sent directly to Teradata. Setting this option to Yes when the SQL statement contains ODBC-specific syntax results in Teradata Database reporting errors.
- **No**: Parsing is enabled in the driver and SQL statements are sent to the parser.

Password | Password used to connect to your database.

RunInQuietMode | Specifies whether the driver should display standard message boxes on the screen.

SessionMode | Specifies the mode for sessions. Values can be Teradata, ANSI, or System Default.
You can enable enhanced ODBC state checking to detect SQLState 24000 (invalid cursor state error) according to the requirements from the ODBC SDK. You have the following options:

- **0**: (Default) No additional state checking is performed and provides backward compatibility. The ODBC driver does not generate error 24000 when an ODBC function generates a result set (open cursor state), and then another function that generates a result set is called before the first results were fetched or freed. This can be caused by ODBC functions such as SQLExecute, SQLExecDirect, or an ODBC catalog function.
- **1**: The driver generates SQLState 24000 error if an application is in an open cursor state and SQLColumns, SQLColumnPrivileges, SQLExecDirect, SQLExecute, SQLForeignKeys, SQLGetTypeInfo, SQLPrepare, SQLPrimaryKeys, SQLProcedureColumns, SQLProcedures, SQLSpecialColumns, SQLStatistics, SQLTablePrivileges, or SQLTables is called.
- **2**: The driver generates SQLState 24000 error if an application is not in an open cursor state and SQLCol, SQLDescribeCol, or SQLFetch is called.

**UseNativeLOBSupport**

Specifies whether native Large Object (LOB) data type mappings are supported. This setting is most relevant for backward compatibility to support databases that do not support native LOB. You have the following options:

- **Yes**: (Default for databases that support LOB) Native LOB is supported for the database. ODBC is used to map the SQL_LONGVARBINARY data type to Teradata's Binary Large Object (BLOB), and to map the SQL_LONGVARCHAR data type to Teradata's Character Large Object (CLOB).
- **No**: Native LOB is not supported for the database. This option should be used for databases that do not support LOB to maintain backward compatibility. ODBC is used to map the SQL_LONGVARBINARY data type to Teradata's VARBYTE(32000), and to map the SQL_LONGVARCHAR data type to Teradata's LONG VARCHAR.

**Username**

User ID used to connect to your database.

---

**Netezza**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>Name of the database to connect.</td>
</tr>
<tr>
<td>Password</td>
<td>Password used to connect to your database.</td>
</tr>
<tr>
<td>Port</td>
<td>The port number for the connection. Check with your database administrator for the correct number. The default is 5480.</td>
</tr>
</tbody>
</table>
| ReadOnly  | Specifies whether intermediate tables can be created.  
  - **0**: (Default) Intermediate tables can be created.  
  - **1**: Intermediate tables cannot be created. |
### Table: Configuring ODBC Parameters with odbc.ini

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServerName</td>
<td>The machine name or IP address of the host machine for the database.</td>
</tr>
<tr>
<td>Username</td>
<td>User ID used to connect to the database.</td>
</tr>
</tbody>
</table>
Introduction

This appendix provides information on common problems that you might encounter while installing and configuring MicroStrategy on UNIX, Linux, and Windows operating systems.

Installing MicroStrategy on UNIX

Some of the problems you can encounter while installing MicroStrategy on UNIX are listed below with steps to resolve them.

For information on application server-specific issues and general troubleshooting for MicroStrategy Web Universal, refer to the MicroStrategy Web Universal release notes.
Mounting and unmounting CD-ROMs

Before you can install MicroStrategy on either the Solaris, AIX, or HP-UX operating system, you need to mount the CD-ROM. The procedure for each is included below, along with instructions to unmount the CD-ROM.

**Mounting and unmounting the CD-ROM on Solaris**

Follow the steps below to mount the CD-ROM on UNIX.

---

**To mount the CD-ROM on Solaris**

1. Enter the command `su` and the root password at the command prompt, or log in as `root` to become the superuser (`su`). The command prompt changes to the pound sign (`#`).

2. If the `/cdrom` directory does not already exist, enter
   ```
   # mkdir /cdrom
   ```

3. If the `/cdrom/cdrom0` directory is empty because the CD-ROM was not mounted, or if File Manager did not open a window displaying the contents of the CD-ROM, verify that the `vold` daemon is running by entering the following:
   ```
   # ps -e | grep vold | grep -v grep
   ```
   The `vold` process manages the CD-ROM device and performs the mounting.

4. If `vold` is running, the system displays the process identification number of `vold`. If the system does not display anything, stop and start the daemon by entering
   ```
   # /usr/sbin/vold &
   ```

5. If the `vold` daemon is running but did not mount the CD-ROM, stop the `vold` daemon and then stop and start it. To stop it you need to know the identification process number, which can be obtained by entering
   ```
   # ps -ef | grep vold | grep -v grep
   ```
6 Stop the vold process by entering

   # kill -15 process_ID_number

7 Stop and start the vold process by entering

   # /usr/sbin/vold &

If you encounter problems using the vold daemon, enter one of the following commands to mount the CD-ROM:

- # etc/mountall
- # mount -F hsfs -r ro /dev/dsk/cxtyd0sz /cdrom/cdrom0

   where \( x \) is the CD-ROM drive controller, \( y \) is the CD-ROM drive SCSI ID number, and \( z \) is the slice of the partition on which the CD-ROM is located.

---

**To unmount the CD-ROM on Solaris**

Type the following commands to unmount the CD-ROM on Solaris:

1   # cd
2   # umount /cdrom/cdrom0
3   # eject

---

**Mounting and unmounting the CD-ROM on AIX**

Follow the steps below to mount the CD-ROM on the AIX UNIX operating system.
To mount the CD-ROM on AIX

1. If necessary, use the `mkdir` command to create a mount point for the CD. The following command creates a mount point at `/cdrom`; you can mount the CD at any location on the machine’s local file system.

   ```
   # mkdir /cdrom
   
   # The commands in this procedure assume the CD is mounted at /cdrom. If you mount the CD at a different location, use that location when issuing commands.
   ```

2. Mount the CD-ROM drive by entering

   ```
   # mount -o ro -v cdrfs /dev/cdnumber /cdrom
   
   In this command, `number` is the CD-ROM number for your system, usually 0 (zero).
   
   An example of the above command follows:
   ```

   ```
   # mount -o ro -v cdrfs /dev/cd0 /cdrom
   ```

3. Navigate to the `/cdrom` directory by entering

   ```
   # cd /cdrom
   
   To unmount the CD-ROM on AIX

   Type the following command to unmount the CD-ROM on AIX:

   ```
   
   # unmount /cdrom
   
   This command assumes the CD is mounted at `/cdrom`. If you mount the CD at a different location, use that location when issuing commands.

   Mounting and unmounting the CD-ROM on HP-UX

   Follow the steps below to mount the CD-ROM on the HP-UX operating system.
To mount the CD-ROM on HP-UX

1. If necessary, use the `mkdir` command to create a mount point for the CD. The following command creates a mount point at `/cdrom`; you can mount the CD at any location on the machine’s local file system.

   ```
   # mkdir /cdrom
   ```

2. To return the CD drive file path, enter

   ```
   # ioscan -funC disk
   ```

3. Mount the CD-ROM drive by entering

   ```
   # mount filepath /cdrom
   ```

   In this command, `filepath` is the CD drive file path returned from step 2.

   An example of the above command follows:

   ```
   # mount /dev/dsk/c0t0d0 /cdrom
   ```

4. Navigate to the `/cdrom` directory by entering

   ```
   # cd /cdrom
   ```

To unmount the CD-ROM on HP-UX

Type the following command to unmount the CD-ROM on HP-UX:

```
# unmount /cdrom
```

This command assumes the CD is mounted at `/cdrom`. If you mount the CD at a different location, use that location when issuing commands.
Reviewing general installation errors

Any errors in your MicroStrategy installation are written to the install.log file.

### Review errors found in install.log

1. **Browse to** INSTALL_PATH **where** INSTALL_PATH **is the directory you specified as the Install Directory during installation**

2. **Open the** install.log **file.** (For Unix, use an editor with a command like dtpad install.log. For Windows, use Notepad or some other text editor to open the file.)

3. **Review the error messages.** A common error is to run out of space.

Graph and document support of non-Western European fonts

If your UNIX system uses non-Western European fonts, you may see indiscernible values returned in place of text on your graphs when accessed through MicroStrategy Web or Web Universal. This also occurs for Report Services documents when accessed through MicroStrategy Web or Web Universal or any other MicroStrategy client application (such as Desktop) that connects to an Intelligence Server connected (three-tier) project source.

To support non-Western European fonts, copy True Type fonts into the Intelligence Server installation directory. Copy these fonts, which have a .ttc or .ttf extension, to INTELLIGENCE_SERVER_INSTALL_PATH\PDFGeneratorFiles. **The default installation path for the Intelligence Server in UNIX is** home\MicroStrategy\PDFGeneratorFiles. **For the change to take effect, you must restart Intelligence Server.**
Server port number errors

This section provides troubleshooting information on server port number errors.

I forgot the server port number

1 Open MicroStrategy Service Manager.
2 Click Options. This launches the Service Options dialog box.
3 Click Intelligence Server Options tab to view the port number.

Port number is in use

In a UNIX or Linux environment, you can find a port available for use with the following procedure:

1 Browse to your target directory. This is the path indicated during installation as the Directory Name.
2 Browse to the folder bin and type mstrctl -s IntelligenceServer di FindingPortNumber, then press ENTER.
3 Type mstrctl -s IntelligenceServer ci FindingPortNumber, then press ENTER.
4 Type mstrctl -s IntelligenceServer gs FindingPortNumber, then press ENTER.
5 An XML file is returned. Search for the tag `<tcp_port_number>`, which contains a port number you can use. Record this number.

**DSN connection errors**

This section provides troubleshooting information on DSN connection errors.

**Testing the DSN connection failed in DSN Creator**

1 In MicroStrategy Connectivity Wizard, go to the Driver Details page and review all the information.

2 Click each of the boxes and read the comments at the bottom of the window.

3 Enter the information required for each box and click **Test**. The connection is either successful, or an error message is displayed to explain the problem.

**Permission errors**

This section provides troubleshooting information on permission errors in a UNIX or Linux environment.

**Missing JVM file**

The installation fails just before it starts transferring files, and the following error is displayed:

```
JVM not found
```
Clear the set group ID on execution(s) bit on the permissions of the directory where the InstallPath for the Intelligence Server is to be placed.

1. In a console window, type:

   
   #chmod g-s directory

   where directory is the InstallPath for Intelligence Server.

2. Press ENTER.
CONFIGURING A WEB.XML FILE TO USE ABSOLUTE PATHS

Introduction

This appendix provides an example of a modified web.xml file in which parameters that originally had relative paths now have absolute paths. The lines that were modified are bolded for easy reference.

Example of modified web.xml file

The following is an example of a modified web.xml file:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems,Inc.//DTD Web Application 2.3//EN" "http://java.sun.com/dtd/web-app_2_3.dtd">
  <web-app>
    <display-name>Web Tier</display-name>
    <context-param>
      <param-name>adminServersFile</param-name>
      <param-value>AdminServers.xml</param-value>
    </context-param>
    <context-param>
      <param-name>dtdLocation</param-name>
```
<param-value>/WEB-INF/xml/dtds</param-value>  
<description>This is the directory where the application looks for DTDs used by Layout Definitions</description>
</context-param>
<context-param>
  <param-name>localesFile</param-name>
  <param-value>/WEB-INF/xml/locales.xml</param-value>
</context-param>
<context-param>
  <param-name>preferencesGroupsFile</param-name>
  <param-value>/WEB-INF/xml/preferencesGroups.xml</param-value>
</context-param>
<context-param>
  <param-name>resourcesFolderImage</param-name>
  <param-value>../images</param-value>
</context-param>
<context-param>
  <param-name>resourcesFolderJavaScript</param-name>
  <param-value>../javascript</param-value>
</context-param>
<context-param>
  <param-name>resourcesFolderJSP</param-name>
  <param-value>../jsp</param-value>
</context-param>
<context-param>
  <param-name>resourcesFolderStyle</param-name>
  <param-value>../style</param-value>
</context-param>
<context-param>
  <param-name>resourcesFolderStyleFixedFont</param-name>
  <param-value>../style/fixed-font</param-value>
</context-param>
<context-param>
  <param-name>resourcesFolderStyleFixedFontDB</param-name>
  <param-value>../style/fixed-font/db</param-value>
</context-param>
<context-param>
  <param-name>serverConfigFilesDefaultLocation</param-name>
  <param-value>ABSOLUTE:/usr/GGapps/MicroStrategy/xml/</param-value>
</context-param>
<context-param>
  <param-name>serverLogFilesDefaultLocation</param-name>
  <param-value>ABSOLUTE:/usr/GGapps/MicroStrategy/xml/log/</param-value>
</context-param>
<context-param>
  <param-name>serverLogPropertiesFile</param-name>
  <param-value>logger.properties</param-value>
</context-param>
</context-param>
<context-param>
    <param-name>styleCatalogFile</param-name>
    <param-value>/WEB-INF/xml/styleCatalog.xml</param-value>
</context-param>
<context-param>
    <param-name>sysPreferenceDefinitionFile</param-name>
    <param-value>sys_definitions.xml</param-value>
</context-param>
<context-param>
    <param-name>sysServerDefaultsFile</param-name>
    <param-value>sys_defaults.xml</param-value>
</context-param>
</context-param>

<!-- Other parameters that are available for customization -->
<context-param>
    <param-name>configServletPath</param-name>
    <param-value /></context-param>
<context-param>
    <param-name>appBeanEventsFile</param-name>
    <param-value></param-value>
</context-param>
<context-param>
    <param-name>mdBeanEventsFile</param-name>
    <param-value></param-value>
</context-param>

<!-- Other servlets -->
<servlet>
    <servlet-name>mstrDocumentImages</servlet-name>
    <servlet-class>com.microstrategy.web.servlets.DocumentImageServlet</servlet-class>
    <init-param>
        <param-name>imagesPath</param-name>
        <param-value>../../images</param-value>
    </init-param>
</servlet>
<servlet>
    <servlet-name>mstrWeb</servlet-name>
    <display-name>Main Servlet</display-name>
    <servlet-class>com.microstrategy.web.servlets.MainServlet</servlet-class>
    <init-param>
        <param-name>adminServletDeploymentName</param-name>
        <param-value>mstrWebAdmin</param-value>
    </init-param>
    <init-param>
        <param-name>blankReportId</param-name>
        <param-value>05B202B9999F4C1AB960DA6208CADF3D</param-value>
    </init-param>
    <init-param>
        <param-name>blankReportWritingId</param-name>
        <param-value></param-value>
    </init-param>
</servlet>
Example of modified web.xml file

```xml
<param-name>clientLoggingFile</param-name>
[param-value>C:\MyTest.log</param-value>
<description>This parameter specifies the location of the log file where the statistics will be saved. The full path should be included as part of the parameter value. If the file does not exist, the logging infrastructure will create it (if possible). If the file already exists, then the new information will be appended to it.</description>

<param-name>clientLoggingForm</param-name>
[param-value>http://localhost:8080/Microstrategy7/jsp/ClientPerformanceForm.jsp</param-value>
<description>This parameter specifies the server and page to be used for displaying the client performance results processing form.</description>

<param-name>clientLoggingLevel</param-name>
[param-value>0</param-value>
<description>This is the level of detail included in the log. Its value is formed by the BITWISE OR-ing of these values:
1=Analyze information at the PAGE (rendering) level,
2=Analyze information at the DETAILED (rendering) level,
4=Analyze information at the METHOD (client manipulation) level,
8=Analyze information at the ACTION (client manipulation) level.
</description>

<param-name>clientLoggingMode</param-name>
[param-value>1</param-value>
<description>This is how the logging is presented to the user. Its value is formed by the BITWISE OR-ing of these values:
1=Log to a file (see the value of clientLoggingFile above for location). For this flag to be applied, it requires the clientLoggingType is set to LIVE.
2=Display results through an HTML page loaded on the client's computer. For this flag to be applied, it requires the clientLoggingType is set to LIVE.
4=Display results through a Browser ALERT box.
</description>
```
```xml
- <init-param>
  <param-name>clientLoggingPage</param-name>
  <param-value>http://localhost:8080/Microstrategy 7/jsp/Client_Log.jsp</param-value>
  <description>This parameter specifies what will be the server and page to be used for displaying client performance results.</description>
</init-param>
- <init-param>
  <param-name>clientLoggingType</param-name>
  <param-value>1</param-value>
  <description>This parameter specifies the type of process that will be used for collecting the statistics information. If the server/page to process the statistical information is going to be available for the client computer and the reporting method involves log file or HTML report, the logging type should be specified to be LIVE (1). If the client computer where the test is going to be performed does not have access to this server, then the logging type is STATIC (2). It does not matter if the pages are being accessed live from the Web Universal application or not.</description>
</init-param>
- <init-param>
  <param-name>configFile</param-name>
  <param-value>/WEB-INF/xml/pageConfig.xml</param-value>
</init-param>
- <init-param>
  <param-name>controllerClass</param-name>
  <param-value>com.microstrategy.web.app.MSTRWebController</param-value>
</init-param>
- <init-param>
  <param-name>defaultPDFImagesFolder</param-name>
  <param-value>/images/</param-value>
</init-param>
- <init-param>
  <param-name>excelDirectExport</param-name>
  <param-value>false</param-value>
</init-param>
- <init-param>
  <param-name>externalSecurityClass</param-name>
  <param-value>com.microstrategy.web.app.DefaultExternalSecurity</param-value>
</init-param>
- <init-param>
  <param-name>formatPropertiesMapping</param-name>
  <param-value>/WEB-INF/xml/formattingProperties.xml</param-value>
</init-param>
```
- <init-param>
  <param-name>iframeJSP</param-name>
  <param-value>/jsp/iframe.jsp</param-value>
</init-param>

- <init-param>
  <param-name>intelligentServerPollingFrequency</param-name>
  <param-value>300000</param-value>
</init-param>

- <init-param>
  <param-name>objBrowserFolderLinksConfigFile</param-name>
  <param-value>/WEB-INF/xml/objBrowserFolderLink.xml</param-value>
</init-param>

- <init-param>
  <param-name>reloadConfigurationFiles</param-name>
  <param-value>false</param-value>
</init-param>

- <init-param>
  <param-name>useServletForPDF</param-name>
  <param-value>true</param-value>
</init-param>

- <servlet>
  <servlet-name>mstrWeb72GraphLoader</servlet-name>
  <servlet-class>com.microstrategy.web.servlets.OldLinksServlet</servlet-class>
</servlet>

- <init-param>
  <param-name>graphBeanName</param-name>
  <param-value>gb</param-value>
</init-param>

- <init-param>
  <param-name>mainServletPath</param-name>
  <param-value>mstrWeb</param-value>
</init-param>

- <init-param>
  <param-name>reportBeanPath</param-name>
  <param-value>mstrWeb.report.frame.rb</param-value>
</init-param>

- <init-param>
  <param-name>resourceName</param-name>
  <param-value>MSIGraphLoader.asp</param-value>
</init-param>

- <servlet>
  <servlet-name>mstrWeb72RebuildReport</servlet-name>
</servlet>
<servlet-class>com.microstrategy.web.servlets.OldLinksServlet</servlet-class>
- <init-param>
  <param-name>mainServletPath</param-name>
  <param-value>mstrWeb</param-value>
</init-param>
- <init-param>
  <param-name>reportBeanPath</param-name>
  <param-value>mstrWeb.report.frame.rb</param-value>
</init-param>
- <init-param>
  <param-name>resourceName</param-name>
  <param-value>RebuildReport.asp</param-value>
</init-param>
</servlet>
- <servlet>
  <servlet-name>mstrWebAdmin</servlet-name>
  <display-name>Main Admin Servlet</display-name>
  <servlet-class>com.microstrategy.web.servlets.AdminServlet</servlet-class>
- <init-param>
  <param-name>adminServletDeploymentName</param-name>
  <param-value>mstrWebAdmin</param-value>
</init-param>
- <init-param>
  <param-name>configFile</param-name>
  <param-value>/WEB-INF/xml/Admin_pageConfig.xml</param-value>
</init-param>
- <init-param>
  <param-name>controllerClass</param-name>
  <param-value>com.microstrategy.web.admin.AdminController</param-value>
</init-param>
- <init-param>
  <param-name>diagnosticsFileMbSizeLimit</param-name>
  <param-value>1</param-value>
</init-param>
- <init-param>
  <param-name>externalSecurityClass</param-name>
  <param-value>com.microstrategy.web.app.DefaultExternalSecurity</param-value>
</init-param>
- <init-param>
  <param-name>servletDeploymentName</param-name>
  <param-value>mstrWebAdmin</param-value>
</init-param>
- <init-param>
  <param-name>webServletDeploymentName</param-name>
  <param-value>mstrWeb</param-value>
</init-param>
Example of modified web.xml file

```xml
<init-param>
    <security-role-ref>
        <role-name>ADMIN</role-name>
        <role-link>admin</role-link>
    </security-role-ref>
</init-param>

<servlet>
    <servlet-mapping>
        <servlet-name>mstrDocumentImages</servlet-name>
        <url-pattern>/servlet/Images/*</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>mstrWeb72GraphLoader</servlet-name>
        <url-pattern>/servlet/MSIGraphLoader.asp</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>mstrWeb72RebuildReport</servlet-name>
        <url-pattern>/servlet/RebuildReport.asp</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>mstrDocumentImages</servlet-name>
        <url-pattern>/servlet/images/*</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>mstrWeb</servlet-name>
        <url-pattern>/servlet/mstrWeb</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>mstrWebAdmin</servlet-name>
        <url-pattern>/servlet/mstrWebAdmin</url-pattern>
    </servlet-mapping>

    <session-config>
        <session-timeout>30</session-timeout>
    </session-config>

    <mime-mapping>
        <extension>css</extension>
        <mime-type>text/css</mime-type>
    </mime-mapping>

    <taglib>
        <taglib-uri>/webUtilTL.tld</taglib-uri>
        <taglib-location>/WEB-INF/tlds/webUtilTL.tld</taglib-location>
    </taglib>

    <security-constraint>
        <web-resource-collection>
            <web-resource-name>Administrator</web-resource-name>
            <url-pattern>/servlet/mstrWebAdmin</url-pattern>
            <http-method>GET</http-method>
        </web-resource-collection>
    </security-constraint>
```
<http-method>POST</http-method>
<http-method>PUT</http-method>
<http-method>DELETE</http-method>
</web-resource-collection>
<auth-constraint>
  <role-name>admin</role-name>
</auth-constraint>
<user-data-constraint>
  <transport-guarantee>NONE
</transport-guarantee>
</user-data-constraint>
</security-constraint>
<login-config>
  <auth-method>BASIC</auth-method>
  <realm-name>Administrator Authentication Area</realm-name>
</login-config>
<security-role>
  <role-name>admin</role-name>
</security-role>
</web-app>
GLOSSARY

**activation code**  A code used to activate MicroStrategy Intelligence Server. This code is sent to an email address provided during activation.

**attribute**  A data level defined by the system architect and associated with one or more columns in a data warehouse lookup table. Attributes include data classifications like Region, Order, Customer, Age, Item, City, and Year. They provide a means for aggregating and filtering at a given level.

**attribute element**  A value of any of the attribute forms of an attribute. For example, New York and Dallas are elements of the attribute City; January, February, and March are elements of the attribute Month.

**caches**  A special data store holding recently accessed information for quick future access. This is normally done for frequently requested reports, whose execution is faster because they need not run against the database. Results from the data warehouse are stored separately and can be used by new job requests that require the same data. In the MicroStrategy environment, when a user runs a report for the first time, the job is submitted to the database for processing. However, if the results of that report are cached, the results can be returned immediately without having to wait for the database to process the job the next time the report is run.
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>connection string</strong></td>
<td>Stores the information required to connect to a database server. A connection string usually includes a DSN and the user ID and password required to log in to the database server. This information varies depending on the particular database server.</td>
</tr>
</tbody>
</table>
| **data source name**  | Provides connectivity to a database through an ODBC driver. A DSN generally contains host machine name or IP address, instance name, database name, directory, database driver, User ID, password, and other information. The exact information included in the DSN varies by DBMS. Once you create a DSN for a particular database, you can use it in an application to call information from the database. There are three types of DSNs:  
  1) System DSN: can be used by anyone who has access to the machine. DSN info is stored in the registry.  
  2) User DSN: is created for a specific user. Also stored in the registry.  
  3) File DSN: DSN information is stored in a text file with .DSN extension. |
| **data warehouse**     | 1) A database, typically very large, containing the historical data of an enterprise. Used for decision support or business intelligence, it organizes data and allows coordinated updates and loads.  
  2) A copy of transaction data specifically structured for query, reporting, and analysis. |
| **document**           | 1. A container for objects representing data coming from one or more reports, as well as positioning and formatting information. A document is used to format data from multiple reports in a single display of presentation quality.  
  2. The MicroStrategy object that supports the functionality defined in (1). |

**DSN** See *data source name (DSN)*.
**hierarchies** A set of attributes defining a meaningful path for element browsing or drilling. The order of the attributes is typically—though not always—defined such that a higher attribute has a one-to-many relationship with its child attributes.

**HTML document** An HTML document is a container for creating dashboards and scorecards to display a group of reports within the MicroStrategy platform. You can use HTML documents to format data from multiple reports and graphs in a single display of presentation quality.

**installation log file** The MicroStrategy setup program generates a log file in text format. This log file contains records of all actions performed by the setup program and by other executable files related to installation.


**metadata** A repository whose data associates the tables and columns of a data warehouse with user-defined attributes and facts to enable the mapping of the business view, terms, and needs to the underlying database structure. Metadata can reside on the same server as the data warehouse or on a different database server. It can even be held in a different RDBMS.

**ODBC** See *Open Database Connectivity*.

**ODBC driver** A software routine that translates MicroStrategy Intelligence Server requests into commands that the DBMS understands.
**ODBC driver manager** Coordinates communication between a client application and database server. The client application tells the driver manager that it needs to connect using a particular connection string. The DSN found in this connection string provides the driver manager with the type of database server to which the application needs access. From this information, the driver manager decides what driver to use and initiates the communication.

**Open Database Connectivity** An open standard with which client computers can communicate with relational database servers. Client machines make a connection to a particular logical database, on a particular physical database server, using a particular ODBC driver.

**port number** The port number is how a server process identifies itself on the machine on which it is running. For example, when the Intelligence Server machine receives a network call from a client (Desktop, Web Universal, Narrowcast Server, Command Manager, and so on), it knows to forward those calls to the Intelligence Server port number that is specified in the call.

**prefix** A prefix is stored in the project metadata associated with a table or tables and is used by the Engine to generate SQL. Also, the Catalog Server uses it to obtain table sample values and row counts. In most cases, it should match the name space field since it is used to qualify on a specific table belonging to a certain owner or name space. Prefixes can be defined and modified from the Warehouse Catalog interface.

**process** An executing application comprising one or more threads. Processes use temporary private address spaces and control operating system resources such as files, dynamic memory allocations, pipes, and synchronization objects.
**project** 1) The highest-level intersection of a data warehouse, metadata repository, and user community, containing reports, filters, metrics, and functions.

2) An object containing the definition of a project, as defined in (1). The project object is specified when requesting the establishment of a session.

**project source** Defines a connection to the metadata database and is used by various MicroStrategy products to access projects. A direct project source is a two-tier connection directly to a metadata repository. A server project source is a three-tier connection to a MicroStrategy Intelligence Server. One project source can contain many projects and the administration tools found at the project source level are used to monitor and administer all projects in the project source.

**schema object** A MicroStrategy object created, usually by a project designer, that relates the information in the logical data model and physical warehouse schema to the MicroStrategy environment. These objects are developed in MicroStrategy Architect, which can be accessed from MicroStrategy Desktop. Schema objects directly reflect the warehouse structure and include attributes, facts, functions, hierarchies, operators, partition mappings, tables, and transformations.

**server definition** A MicroStrategy object stored in the metadata containing information about the configuration of an Intelligence Server.

**server instance** The combination of an Intelligence Server running with a particular server definition.

**statistics tables** Tables that are used to record a variety of statistical information about the usage and performance of a MicroStrategy system.

**Structured Query Language (SQL)** The query language standardized in 1986 by the American National Standards Institute (ANSI) and used to request information from tables in a relational database and to manipulate the tables’ structure and data.
absolute path, web.xml 535
Access. See Microsoft Access.
accessing
  MicroStrategy Connectivity Wizard 152
  MicroStrategy Web Administrator page for Sun Java System 254
  MicroStrategy Web Services Home page for Sun Java System 317
activating installation 134
  activation code 132
  FAQ 135
  requesting activation code on UNIX 124
  requesting activation code on Windows 100
activation code defined on 132
  activating installation with 134
  requesting 132
  UNIX 124
  Windows 100
adding/removing MicroStrategy components (Windows) 456
admin role, assigning to a user
  JBoss - Windows - Web/Web
  Universal 284
  Oracle 10g - Web Services 330
  Oracle 10g - Windows - Web/Web Universal 278
  Sun Java System - Solaris - Web/Web Universal 252
  Sun Java System - Web Services 316
  Tomcat - Linux 268
  Tomcat - Windows - Web/Web Universal 262
  WebSphere - AIX - Web/Web Universal 241
  WebSphere - Web Services 309
Administrative Server 237
AIX
  MicroStrategy Intelligence Server
    hardware requirements 29
    patch requirements 56
    Web server OS 60
Analytics Modules. See MicroStrategy Analytics Modules.
Apache Tomcat application server 61
Apache Web server
  configuring to serve a static file 232
  deploying MicroStrategy Mobile Server J2EE with 216
SequeLink ODBC Socket Server 17

configuring
  Apache Web server to serve a static file 232
  command line mode (UNIX) 411, 424, 427, 430
  JBoss 282
  metadata repository and statistics tables 162
  MicroStrategy command line mode 431
  Sun JDK 258, 281
  Sun JDK (Tomcat) 265
  Tomcat 259, 266

configuring ODBC
  default directory (UNIX) 150
  default directory (Windows) 149
  UNIX 150
  Windows 149
  connection errors, data source name 532
  connection mapping, creating 206
  connection string defined on 147
  connectivity 155
  Connectivity Wizard. See MicroStrategy Connectivity Wizard.

D

  data source name 146
    connecting to 164
    connection errors 532
    creating with Connectivity Wizard 151
    creating with Microsoft ODBC Administrator 154
    data warehouse 200
    data source, accessing 193
    data warehouse defined on 146
    communicating with 144

  connecting to 192
  connection setting 193
  DSN 200
  login 204
  database authentication 183
  database connection defined on 200
  certified ODBC driver 79
  database instance defined on 193
  creating 199
  database login defined on 204
  creating 205
  database management system (DBMS) defined on 144
  DB Query Tool 155
  DB2 for i
    ODBC driver for 468
  DB2 for UDB iSeries
    ODBC driver for 468
  DB2 UDB, ODBC driver for
    (UNIX/Linux) 467
  DB2 Wire Protocol
    parameters for odbc.ini 504
  DB2 Wire Protocol for UDB
    ODBC driver for (Windows) 467
  DB2 z/OS
    ODBC driver for 469
  DBMS. See database management system.
  deploying MicroStrategy Mobile Server
    J2EE 220
    JBoss (Windows) 280
    manually deploying with
      WebLogic 220
    Oracle 10g 275
    SAP NetWeaver (Windows) 270
    Sun Java System 246
    Tomcat 257
    WAR file with Sun Java System 251
    WebLogic and Apache 216
    WebSphere and IBM HTTP Server 234
deploying MicroStrategy Mobile Server with ASP.NET and IIS 212
deploying MicroStrategy Web 211
deploying MicroStrategy Web Services J2EE automatically 294
automatically deploying with WebLogic 294
manually deploying with WebLogic 294
Oracle 10g 328
Sun Java System 311, 313
Tomcat 321
WAR file (Sun Java System) 315
WebLogic and Apache 295
WebSphere 303, 325
WebSphere and IBM HTTP Server 301
Windows 321
deploying MicroStrategy Web Universal automatically deploying with WebLogic 220
JBoss (Windows) 280
Linux 268
manually deploying with WebLogic 220
Oracle 10g 275, 277
SAP NetWeaver (Windows) 270, 272
Sun Java System 246, 249
Tomcat 257
Tomcat (Linux) 268
WAR file with Sun Java System 251
WebLogic and Apache 216, 220
WebSphere 236
WebSphere and IBM HTTP Server 234
direct project source 178, 179
document defined on 336
driver. See ODBC driver.
DSN Creator 532

element browsing defined on 72
Enterprise Manager. See MicroStrategy Enterprise Manager.
example
response.ini file for installation (Windows) 366
response.ini file for MicroStrategy Configuration Wizard (Windows) 184
response.ini file for uninstallation 397
web.xml 535
Excel. See Microsoft Excel.

file
install.log 347
preinst.log 399
setu.exe 346
folder structure for WebLogic and Apache 216
Freeform SQL, accessing data sources 193

Hardware recommendations
Web Universal on UNIX/Linux 32
Windows 24
hardware requirements for Intelligence
Server on UNIX/Linux 27

history list
  creating repository 167
tables 167

hotfix, silent installation (Windows) 401
HP-UX
  Java development kit 64
  MicroStrategy Intelligence Server hardware requirements 30
  patch requirements 57
  Web server OS 60
HTML documents 335, defined on 337
  setting up document creation 338
  setting up document execution 341

I

IBM HTTP Server
  configuring to serve a static file 246
  web server 62
IBM JDK, Java development kit 64
IBM WebSphere, application server 61
IBM, Java development kit 64
IIS. See Microsoft IIS.
Informix 8
  ODBC driver for (UNIX/Linux) 485
  ODBC driver for (Windows) 471
Informix Wire Protocol
  ODBC Driver for 469, 470
  parameters for odbc.ini 510
Informix XPS, parameters for odbc.ini 508
install.log 347
installation
  creating response.ini file (Windows) 349
  install.log file 347

OEM silent (Windows) 418
plug-in for Apache 219
response.ini file (Windows) 348
silent (UNIX) 406
silent (Windows) 398
silent installation of hotfix (Windows) 401
troubleshooting silent (Windows) 400

installation log file 346, defined on 346
installation method 84
  Command line mode 85
  GUI mode 84
  silent installation 85
installation prerequisites 19
  hardware recommendations for Web Universal on UNIX/Linux 32
  hardware recommendations for Windows 24
  hardware requirements for Intelligence Server on UNIX/Linux 27
  large production deployment 23
  recommended installation location 20
  small production deployment 21
  software recommendations for Web Services J2EE 64
  software recommendations for Web Universal 58
  software recommendations for Windows 35
  software requirements for Intelligence Server on UNIX/Linux 52
installation procedure
checklist 87
Linux 112
MicroStrategy Installation Wizard on Linux 115
MicroStrategy Installation Wizard on UNIX 115
MicroStrategy Installation Wizard on Windows 93
UNIX 112
Windows 92
Intelligence Server. See MicroStrategy Intelligence Server.
Intelligent Cube size 78
international support xxvii, 78

J
Java development kit
IBM JDK 64
Sun JDK 64
supported 64
Java heap size, setting
Sun Java System (Web Services) 319
Sun Java System (Web/Web Universal) 256
WebLogic and Apache (Web Services) 301
WebLogic and Apache (Web/Web Universal) 229
WebSphere (Web Services) 310
WebSphere (Web/Web Universal) 244
JBoss (Web/Web Universal)
configuring 282
JBoss (Windows - Web/Web Universal)
preconfiguration information 281
JSP file, precompiling 229, 245

L
launching the project on WebLogic and Apache 228
launching WebSphere Administrative Console
MicroStrategy Web Services 304
Web/Web Universal 237
LDAP
authentication 183
license 86

Linux
deploying MicroStrategy Web Universal 268
installation procedure 112
MicroStrategy Installation Wizard 115
MicroStrategy Intelligence Server hardware requirements 31
patch requirements 57
Web server OS 60
log
installation 346
statistics 74

M
manual deployment
MicroStrategy Mobile Server
J2EE 220
Web Services J2EE 294
Web Universal 220
MDX cube sources
accessing data sources 193
installing the TM1 Connector 105
metadata defined on 146, 162
table 164
metadata repository 158, 162, 174, 416
creating 164
methods of automated installation
(Windows) 347
Microsoft Access
connecting as data source
(UNIX/Linux) 475
preparing database for access 476
Microsoft Excel
connecting as data source 494
connecting as data source
(UNIX/Linux) 475
creating DSN 495
preparing as data source 494
preparing file for access 477
Microsoft IIS 70, 210
ASP.NET and MicroStrategy Mobile Server 212
ASP.NET and MicroStrategy Web 211
supporting IIS 7 or 7.5 68
web server 62
Microsoft ODBC Administrator 154
creating a DSN using 154
Microsoft SQL Server, ODBC driver for 474
MicroStrategy
uninstalling (UNIX/Linux) 461
MicroStrategy Administrator
MicroStrategy Command Manager 4
MicroStrategy Enterprise Manager 5
MicroStrategy Object Manager 4
MicroStrategy Analytics Modules 18
MicroStrategy Architect 3
MicroStrategy Command Manager 4
MicroStrategy common setups 20
MicroStrategy component 2
MicroStrategy Analytics Modules 18
MicroStrategy Desktop 3
MicroStrategy Health Center 19
MicroStrategy Intelligence Server 5
MicroStrategy Narrowcast Server 17
MicroStrategy Office 15
MicroStrategy Portlets 10
MicroStrategy SDK 13
MicroStrategy Tutorial Reporting 18
MicroStrategy Web and Web Universal 8
MicroStrategy Web MMT 10
MicroStrategy Web Services 16, 88
re-installing 457
re-installing MicroStrategy Office
MicroStrategy Office re-installing 458
SequeLink ODBC Socket Server 17
MicroStrategy Configuration Wizard
accessing on UNIX/Linux 161
accessing on Windows 160
connecting to a DSN 164
creating a metadata repository 164
overview 159
prerequisites 158
response.ini file (Windows) 183, 396
setting up Intelligence server 173
MicroStrategy Connectivity Wizard 151
accessing 152
creating a DSN 151
MicroStrategy Desktop
MicroStrategy Architect 3
MicroStrategy Desktop Analyst 3
MicroStrategy Desktop Designer 3
MicroStrategy Function Plug-in Wizard 3
MicroStrategy Project Builder 3
MicroStrategy Server Administrator 4
subcomponents 3
MicroStrategy Desktop Analyst 3
MicroStrategy Desktop Designer 3
MicroStrategy Distribution Services 7
MicroStrategy Enterprise Manager 5, 170
MicroStrategy Function Plug-in Wizard 3
MicroStrategy Health Center 19
MicroStrategy installation
prerequisites 19
MicroStrategy Installation Wizard
accessing on Linux 115
accessing on UNIX 115
accessing on Windows 93
MicroStrategy Intelligence Server 5, 163, 176
certified database and ODBC driver 79
compatibility with Web Universal 79
Index

connecting to Mobile Server 213
connecting to Web 211
creating a server definition 175
database connection 144
Enterprise edition 6
hardware requirements (UNIX/Linux) 27
MicroStrategy Distribution Services 7
MicroStrategy OLAP Services 7
MicroStrategy Report Services 7
migrating from Windows to UNIX/Linux 125
selecting a project 176
setting up 173
software requirements (UNIX/Linux) 52
Standard edition 6
subcomponents 6
Universal edition 6
MicroStrategy Mobile
supporting on IIS 7 or 7.5 68
MicroStrategy Mobile Server
connecting to Intelligence Server 213
deploying with ASP.NET and IIS 212
MicroStrategy Mobile Server Administrator page
controlling access to (JBoss Windows) 284
controlling access to (Tomcat Linux) 269
controlling access to (Tomcat Windows) 262
MicroStrategy Mobile Server Administrator page, configuring for WebSphere 242
MicroStrategy Mobile Server J2EE
deploying 216
deploying manually with WebLogic 220
deploying with JBoss (Windows) 280
deploying with SAP NetWeaver (Windows) 270
MicroStrategy Narrowcast Server 17
MicroStrategy Object Manager 4
MicroStrategy Office
allowing a user to install from a network location 16
allowing installation from MicroStrategy Web 97
MicroStrategy OLAP Services 7
sizing guidelines 73
MicroStrategy Portlets 10
MicroStrategy Professional Services 75
MicroStrategy Project Builder 3
MicroStrategy Report Services 7
setting up document execution 343
MicroStrategy Server Administrator 4
MicroStrategy Tutorial Reporting 18
MicroStrategy Web 209
allowing a user to install MicroStrategy Office from 97
ASP.NET 211
connecting to Intelligence Server 211
deploying 211
supporting on IIS 7 or 7.5 68
Web Analyst 9
Web Professional 9
Web Reporter 9
MicroStrategy Web Administrator page 286
accessing (JBoss Windows) 285
accessing (Oracle) 279
accessing (Tomcat Linux) 269
accessing (Tomcat Windows) 263
controlling access to (JBoss Windows) 284
controlling access to (Sun Java System) 252
controlling access to (Tomcat
Linux) 269
controlling access to (Tomcat Windows) 262
MicroStrategy Web and Web Universal versions 9
MicroStrategy Web MMT 10
MicroStrategy Web Services 290
MicroStrategy Web Services home page
accessing (Oracle) 330
accessing (Tomcat) 323
controlling access to (Sun Java System) 315
MicroStrategy Web Services J2EE manual deployment 294
undeploying 319
MicroStrategy Web Universal 9
compatibility with Intelligence Server 79
deploying automatically with WebLogic 220
deploying manually with WebLogic 220
deploying on Oracle 10g 277
deploying with JBoss (Windows) 280
deploying with SAP NetWeaver (Windows) 270
launching project (JBoss Windows) 286
launching project (Oracle 10g) 280
launching project (Tomcat Linux) 270
launching project (Tomcat Windows) 264
undeploying 256
Web Analyst 9
Web Professional 9
Web Reporter 9
migrating MicroStrategy Intelligence Server 125
MSTRInst.log 399
MultiSource Option 193
MySQL
ODBC driver for (Linux) 488
Wire Protocol driver for 471
N
Netezza
ODBC driver for (UNIX/Linux) 486
ODBC driver for (Windows) 475
parameters for odbc.ini 522
network bandwidth 77
O
Object Manager. See MicroStrategy Object Manager.
ODBC defined on 144, 146
default directory (UNIX) 150
default directory (Windows) 149
parameters for odbc.ini 518
ODBC connectivity 155
ODBC driver defined on 147
Aster (Linux) 490
certified 148
certified database 79
DB Query Tool 155
DB2 for i 468
DB2 for UDB iSeries 468
DB2 UDB (UNIX/Linux) 467
DB2 UDB (Windows) 467
DB2 z/OS 469
Informix 8 (UNIX/Linux) 485
Informix 8 (Windows) 471
Informix Wire Protocol 469, 470
Microsoft SQL Server 474
MySQL (Linux) 488
MySQL Wire Protocol 471
Netezza (UNIX/Linux) 486
Netezza (Windows) 475
odbc.ini file (UNIX/Linux) 499
Oracle Wire Protocol 472
other relational databases for Windows 493
PostgreSQL Wire Protocol 473
Red Brick (Linux) 480
Sequelink 475
Sybase Adaptive Server IQ (UNIX/Linux) 482
Sybase ASE Wire Protocol 473
Teradata (UNIX/Linux) 483
Teradata (Windows) 493
testing connectivity 155
text files 497
Vertica (Linux) 492
ODBC driver manager defined on 147
odbc.ini 499
  DB2 Wire Protocol 504
  Informix Wire Protocol 510
  Informix XPS 508
Netezza 522
ODBC 518
Oracle Wire Protocol 501
Red Brick 519
Sybase Adaptive Server IQ 519
Sybase ASE Wire Protocol 511
Teradata 520
Open Database Connectivity defined on 144
options file 406
Oracle 10g
  application server 61
deploying MicroStrategy Mobile Server J2EE 275
deploying MicroStrategy Web Services J2EE 328
deploying MicroStrategy Web Universal 275, 277
deploying Web Services with the Oracle Enterprise Manager 329
deploying Web Universal with the Oracle Enterprise Manager 277
locating WAR file (MicroStrategy Mobile Server J2EE) 277
locating WAR file (Web Services) 327
locating WAR file (Web/Web Universal) 276
preconfiguration information (Web Services) 327
preconfiguration information (Web/Web Universal) 275
Oracle iPlanet
  Web server 62
Oracle WebLogic application server 61
Oracle Wire Protocol
  ODBC driver for 472
  parameters for odbc.ini 501
P
  pageCheckSeconds parameter 231
parameters
  DB2 Wire Protocol 504
  Informix Wire Protocol 510
  Oracle Wire Protocol 501
  Red Brick 519
password to metadata 174
patch requirements
  AIX 56
  HP-UX 57
  Linux 57
  Solaris 55
PDF configuration, Sun Java System
  Web Services 313
  Web/Web Universal 249
performance-based setup
MicroStrategy Professional Services 75
WebLogic (Web Services) 300
WebLogic and Apache (Web/Web Universal) 229
permission errors 532
plug-in for Apache 219
plugin-cfg.xml, regenerating 240
port number defined on 180
Intelligence Server 176
portlets. See MicroStrategy Portlets.
PostgreSQL Wire Protocol
ODBC driver for 473
precompiling
JSP file in WebLogic 229
JSP file in WebSphere 245
preconfiguration information
Apache (Web/Web Universal) 217
JBoss (Windows - Web/Web Universal) 281
Oracle 10g (Web Services) 327
Oracle 10g (Web/Web Universal) 275
Sun Java System (Web Services) 312
Sun Java System (Web/Web Universal) 247
Tomcat (Linux - Web/Web Universal) 264
Tomcat (Web Services) 321
Tomcat (Windows - Web/Web Universal) 257
WebLogic (Web Services) 293
WebLogic (Web/Web Universal) 217
WebSphere (Web Services) 302
WebSphere (Web/Web Universal) 234
preinst.log 399
prerequisites for MicroStrategy Configuration Wizard 158
processor speed 76
production deployment
large 23
small 21
project 193
accessing multiple data sources 193
Project Builder. See MicroStrategy Project Builder.
project source 162, 163
authentication 181
direct 178, 179
server 178
three-tier 180
two-tier 179
Q
Query Builder, accessing data sources 193
R
Red Brick
ODBC driver for 480
parameters for odbc.ini 519
re-deploying
Web Services with WebLogic 300
re-deploying Web Universal with WebLogic 228
re-installing a MicroStrategy component (Windows) 457
control panel 457
re-installing MicroStrategy Office 458
requesting activation code 132
response.ini file
creating for installation (Windows) 349
creating through MicroStrategy Configuration Wizard (Windows) 184
example for installation
(Windows) 366
example for uninstall (Windows) 397
installation (Windows) 348
MicroStrategy Configuration Wizard
(Windows) 183, 396
parameters and options for Configuration
Wizard (Windows) 186
parameters and options for installation
(Windows) 349
uninstalling (Windows) 396
using for Configuration Wizard
(Windows) 186
using for installation (Windows) 395
using through Configuration Wizard
(UNIX) 448

restarting application server
Sun Java System (Web Services) 318
Sun Java System (Web/Web
Universal) 254
restarting WebSphere application server
Web Services 307
Web/Web Universal 240

S
SAP NetWeaver (Windows)
deploying MicroStrategy Web
Universal 272
SAP NetWeaver, locating WAR file 271
SAP NetWeaver, locating Web Services
WAR file 324
SDK. See MicroStrategy SDK.
security
access to MicroStrategy Web Adminis-
trator page 262, 269, 284
access to MicroStrategy Web Adminis-
trator page Sun Java
System 252
access to MicroStrategy Web Services
Home page Sun Java
System 315

SequeLink
configuring ODBC driver 479
Sequelink
ODBC driver for 475
SequeLink ODBC Socket Server 17
Server Administrator. See MicroStrategy
Server Administrator.
server definition defined on 163, 175
connectivity setting 192
server project source 178
setting up document creation
HTML documents 338
setting up document execution
HTML documents 341
Report Services documents 343
setup command-line parameter
(Windows) 395
setup.exe 346
silent installation 398
hotfix (Windows) 401
introduction (UNIX) 406
OEM (Windows) 418
options file (UNIX) 406
output (UNIX) 410
output (Windows) 399
override options 410
perform on Windows 418
settings (UNIX) 407
System Management Server (SMS) on
Windows 399
troubleshooting (Windows) 400
values for different languages 421
sizing guidelines 71
caching 72
Intelligent Cube size 78
MicroStrategy OLAP Services 73
MicroStrategy Professional
Services 75
network bandwidth 77
number of users 71
processor speed 76
report complexity 71
Report Services document 73
SMS, See System Management Server.
software recommendations
application server 61
Java development kit 64
Web Services J2EE 64
Web Universal 58
Windows 35
software requirements for Intelligence Server on UNIX/Linux 52
Solaris
MicroStrategy Intelligence Server
hardware requirements 28
patch requirements 55
Web server OS 60
specifying a port number 176
SQL
query 157
script 172
starting
Administrative Server 237
Apache Web Server 243
Sun Java System Administration Tool 250
Sun Java System Administration Tool (Web Services) 314
Web module for WebSphere 241, 308
WebSphere Administrative Console 237, 304
WebSphere default application server 238, 304
static file
configuring Apache Web server for 232
configuring IBM HTTP Server 246
statistics
log 74
table 170
statistics table defined on 163
Sun Java development kit 64
Sun Java System
application server 61
deploying MicroStrategy Mobile Server J2EE 246
deploying MicroStrategy Web Universal 246
deploying MicroStrategy Web Universal as a WAR file 251
locating WAR file (MicroStrategy Mobile Server J2EE) 249
recommended configuration for UNIX/Linux 247
Sun Java System (Web Services)
accessing MicroStrategy Web Services Home page 317
deploying MicroStrategy Web Services J2EE 311, 313
deploying MicroStrategy Web Services J2EE as a WAR file 315
locating WAR file 312
PDF configuration 313
preconfiguration information 312
recommended configuration for UNIX/Linux 312
restarting application server 318
setting Java heap size 319
undeploying Web Services J2EE 319
Sun Java System (Web/Web Universal)
accessing MicroStrategy Web Administrator page 254
connecting to the Project page 255
deploying MicroStrategy Mobile Server J2EE as a WAR file 251
deploying MicroStrategy Web Universal 249
locating WAR file 248
PDF configuration 249
preconfiguration information 247
recommended configuration for UNIX/Linux 247
recommended configuration for Windows 248
restarting application server 254
setting Java heap size 256
undeploying Web Universal 256
Sun Java System Admin Console
launching 314
launching (Web/Web Universal) 250
Sun JDK
configuring 258, 281
support
international xxvii, 78
support. See technical support.
supporting IIS 6 70
Sybase Adaptive Server IQ
ODBC driver for (UNIX/Linux) 482
parameters for odbc.ini 519
Sybase ASE Wire Protocol
ODBC driver for 473
parameters for odbc.ini 511
System Management Server (SMS) on Windows silent installation 399

T
table, metadata 164
technical support xxviii
Teradata
ODBC driver for (UNIX/Linux) 483
ODBC driver for (Windows) 493
parameters for odbc.ini 520
text file
ODBC driver for (Windows) 497
three-tier project source 180
TM1 Connector for MicroStrategy
installing 105
Tomcat
deploying MicroStrategy Mobile Server J2EE 257
deploying MicroStrategy Web Universal 257
Tomcat (Linux - Web/Web Universal)
deploying MicroStrategy Web Universal 268
locating the WAR file 267
preconfiguration information 264
Tomcat (Web Services)
deploying MicroStrategy Web Services J2EE 321
preconfiguration information 321
Tomcat (Web/Web Universal)
configuring 259, 266
configuring Sun JDK 265
Tomcat (Windows - Web/Web Universal)
preconfiguration information 257
troubleshooting
DSN connection errors 532
permission errors 532
server port number errors 531
UNIX 525
Tutorial Reporting. See MicroStrategy Tutorial Reporting.
two-tier project source 179

U
unattended installation (UNIX) 406
undeploying
Web Services J2EE using command line 319
uninstalling
Control Panel (Windows) 459
MicroStrategy (UNIX/Linux) 461
response.ini (Windows) 396
uninstalling MicroStrategy components
  (Windows) 459
  MicroStrategy Office (Windows) 460
UNIX
  configuring ODBC 150
  installation procedure 112
  MicroStrategy Installation Wizard 115
  verifying installation 127
user assigned to admin role
  JBoss - Windows (Web/Web Universal) 284
  Oracle 10g - Windows (Web/Web Universal) 278
  Oracle 10g (Web Services) 330
  Sun Java System - Solaris (Web/Web Universal) 252
  Sun Java System (Web Services) 316
  Tomcat - Linux 268
  Tomcat - Windows (Web/Web Universal) 262
  WebSphere - AIX- Web/Web Universal) 241
  WebSphere (Web Services) 309
user connection mapping. See connection mapping.

V
verifying installation
  UNIX 127
  Windows 108
  Vertica, ODBC driver for (Linux) 492

W
WAR file
  locating for JBoss (Windows) (MicroStrategy Mobile Server J2EE) 283
  locating for Oracle 10g (MicroStrategy Mobile Server J2EE) 277
  locating for SAP NetWeaver (MicroStrategy Mobile Server J2EE) 272
  locating for Sun Java System (MicroStrategy Mobile Server J2EE) 249
  locating for Tomcat (Linux) (MicroStrategy Mobile Server J2EE) 267
  locating for WebLogic (MicroStrategy Mobile Server J2EE) 219
  locating for WebSphere (MicroStrategy Mobile Server J2EE) 236
WAR file (Web Services)
  exploding for WebLogic 296
  locating for Oracle 10g 327
  locating for SAP NetWeaver 324
  locating for Sun Java System 312
  locating for Tomcat 321
  locating for WebLogic 294
  locating for WebSphere 302
  unzipping for WebLogic and Apache 297
WAR file (Web/Web Universal)
  exploding for WebLogic and Apache (Web/Web Universal) 223
  locating for JBoss (Windows) 282
  locating for Oracle 10g 276
  locating for SAP NetWeaver 271
  locating for Sun Java System 248
  locating for Tomcat (Linux) 267
  locating for Tomcat (Windows) 260
  locating for WebLogic 218
  locating for WebSphere 235
  unzipping for WebLogic and Apache (Web/Web Universal) 225
Web MMT. See MicroStrategy Web MMT.
Web module for WebSphere, starting 241, 308
Web server
Apache 62
IBM HTTP Server 62
Internet Information Server 62
Oracle iPlanet 62
Web server OS
AIX 60
HP-UX 60
Linux 60
Solaris 60
Windows 60
Web Universal
connection 286, 333
hardware recommendations for UNIX/Linux 32
software recommendations 58
Web Universal. See also MicroStrategy Web Universal.
web.xml
absolute path 535
example 535
re-deploying the application 228, 300
sample modified 535
WebLogic (Solaris - Web/Web Universal) 216
WebLogic (Web Services)
deploying MicroStrategy Web Services J2EE 295
exploding the WAR file 296
performance-based setup 300
preconfiguration information 293
production mode 294
WebLogic and Apache
deploying MicroStrategy Mobile Server J2EE 216
locating the WAR file (MicroStrategy Mobile Server J2EE) 219
production mode 220
WAR file 225
WebLogic and Apache (Web Services)
locating the WAR file 294
setting Java heap size 301
WAR file 297
WebLogic and Apache (Web/Web Universal)
deploying MicroStrategy Web Universal 216, 220
exploding the WAR file 223
folder structure 216
launching the project 228
locating the WAR file 218
pageCheckSeconds parameter 231
performance-based setup 229
precompiling JSP file 229
preconfiguration information 217
setting Java heap size 229
WebSphere
deploying MicroStrategy Web Services J2EE 303
locating the WAR file (MicroStrategy Mobile) 236
WebSphere (Web Services) 301
application installation 305
deploying MicroStrategy Web Services J2EE 325
launching Administrative Console 304
locating the WAR file 302
preconfiguration 302
restarting application server 307
setting Java heap size 310
starting default application server 304
WebSphere (Web/Web Universal) 234
application installation 238
deploying MicroStrategy Web Universal 236
launching Administrative Console 237
locating the WAR file 235
precompiling the JSP file 245
preconfiguration 234
regenerating plugin-cfg.xml 240
restarting application server 240
setting Java heap size 244
starting default application server 238

WebSphere and IBM HTTP Server
deploying MicroStrategy Web
Universal 234

WebSphere and IBM HTTP Server, deploying MicroStrategy Web Services
J2EE 301

Windows
authentication 182
configuring ODBC 149
deploying MicroStrategy Web Services
J2EE 321

hardware recommendations 24
installation procedure 92
MicroStrategy Installation Wizard 93
software recommendations 35
supporting IIS 6 70
verifying installation 108
Web server OS 60

Windows INI 348

X
XSL report style 74