

Feature Data Objects (FDO)

# The Essential FDO

The Autodesk logo is displayed in white text on a black rectangular background. The word "Autodesk" is written in a sans-serif font, with a registered trademark symbol (®) at the end. The logo is oriented vertically, reading from bottom to top.

Autodesk®

March 2008

© 2008 Autodesk, Inc. All Rights Reserved. Except as otherwise permitted by Autodesk, Inc., this publication, or parts thereof, may not be reproduced in any form, by any method, for any purpose.

Certain materials included in this publication are reprinted with the permission of the copyright holder.

#### **Trademarks**

The following are registered trademarks or trademarks of Autodesk, Inc., in the USA and other countries: 3DEC (design/logo), 3December, 3December.com, 3ds Max, ActiveShapes, Actrix, ADI, Alias, Alias (swirl design/logo), AliasStudio, AliasWavefront (design/logo), ATC, AUGI, AutoCAD, AutoCAD Learning Assistance, AutoCAD LT, AutoCAD Simulator, AutoCAD SQL Extension, AutoCAD SQL Interface, Autodesk, Autodesk Envision, Autodesk Insight, Autodesk Intent, Autodesk Inventor, Autodesk Map, Autodesk MapGuide, Autodesk Streamline, AutoLISP, AutoSnap, AutoSketch, AutoTrack, Backdraft, Built with ObjectARX (logo), Burn, Buzzsaw, CAiCE, Can You Imagine, Character Studio, Cinestream, Civil 3D, Cleaner, Cleaner Central, ClearScale, Colour Warper, Combustion, Communication Specification, Constructware, Content Explorer, Create>what's>Next> (design/logo), Dancing Baby (image), DesignCenter, Design Doctor, Designer's Toolkit, DesignKids, DesignProf, DesignServer, DesignStudio, DesignStudio (design/logo), Design Your World, Design Your World (design/logo), DWF, DWG, DWG (logo), DWG TrueConvert, DWG TrueView, DXF, EditDV, Education by Design, Exposure, Extending the Design Team, FBX, Filmbox, FMDesktop, Freewheel, GDX Driver, Gmax, Heads-up Design, Heidi, HOOPS, HumanIK, i-drop, iMOUT, Incinerator, IntroDV, Inventor, Inventor LT, Kaydara, Kaydara (design/logo), LocationLogic, Lustre, Maya, Mechanical Desktop, MotionBuilder, Mudbox, NavisWorks, ObjectARX, ObjectDBX, Open Reality, Opticore, Opticore Opus, PolarSnap, PortfolioWall, Powered with Autodesk Technology, Productstream, ProjectPoint, ProMaterials, Reactor, RealDWG, Real-time Roto, Recognize, Render Queue, Reveal, Revit, Showcase, ShowMotion, SketchBook, SteeringWheels, StudioTools, Topobase, Toxik, ViewCube, Visual, Visual Bridge, Visual Construction, Visual Drainage, Visual Hydro, Visual Landscape, Visual Roads, Visual Survey, Visual Syllabus, Visual Toolbox, Visual Tugboat, Visual LISP, Voice Reality, Volo, Wiretap, and WiretapCentral

The following are registered trademarks or trademarks of Autodesk Canada Co. in the USA and/or Canada and other countries: Backburner, Discreet, Fire, Flame, Flint, Frost, Inferno, Multi-Master Editing, River, Smoke, Sparks, Stone, and Wire

All other brand names, product names or trademarks belong to their respective holders.

#### **Disclaimer**

THIS PUBLICATION AND THE INFORMATION CONTAINED HEREIN IS MADE AVAILABLE BY AUTODESK, INC. "AS IS." AUTODESK, INC. DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE REGARDING THESE MATERIALS.

Published By: Autodesk, Inc.  
111 McInnis Parkway  
San Rafael, CA 94903, USA

# Contents

<b>Chapter 1</b>	<b>The Essential FDO</b>	<b>1</b>
	What Is the Feature Data Objects API?	1
	Objectives	3
	Related Documentation	3
	Provider Overview	5
	Feature Description	7
	System Requirements	9
	Operating System and RDBMS Client Software	9
	Development Environment Software	9
	Windows XP, SP2	10
	Red Hat Enterprise Linux 3	10
	Getting Started	10
	Install FDO	10
	Open Source Edition	10
	Enterprise Edition	10
	Setup the C++ Application Development Environment	11
	Windows	11
	Linux	12
	Access the C++ API Reference Documentation	13
	Write the Code to Connect to a Provider	13
	Create a Provider Registry	13
	Create a Connection Manager	14
	Create a Connection	14
	Get the Provider Data Store Type	14

	Get the List of Dependent Files . . . . .	14
	Get the Connection Properties . . . . .	15
	Get Values for the Connection Properties . . . . .	15
	Set the Connection Properties . . . . .	16
	Open a Connection . . . . .	16
	Open a Pending Connection . . . . .	16
	Get the List of Dependent Files . . . . .	17
	Next Steps . . . . .	17
	ArcSDE Provider . . . . .	17
	MySQL Provider . . . . .	17
	Capabilities API . . . . .	17
	Command Capabilities . . . . .	18
	User Management API . . . . .	18
	User Management Capabilities . . . . .	18
	Data Store Management API . . . . .	19
	Spatial Context API . . . . .	21
	Glossary . . . . .	21
	Tolerance . . . . .	21
<b>Chapter 2</b>	<b>The Enterprise Providers . . . . .</b>	<b>23</b>
	System Requirements . . . . .	23
	Software . . . . .	23
	Disk Space and Memory . . . . .	25
	Windows . . . . .	25
	Linux . . . . .	26
	Feature Matrix . . . . .	26
	Connection API . . . . .	28
	File Formats Supported by Raster Provider . . . . .	31
	Command Capabilities . . . . .	32
	User Management API . . . . .	35
	User Management SQL Commands . . . . .	35
	Data Store Management API . . . . .	41
	Spatial Context API . . . . .	43
	Exception Messages . . . . .	44
	Connection Not Established . . . . .	44
	Membership of the public role cannot be changed . . . . .	45
<b>Chapter 3</b>	<b>Installing RDBMS Clients Used By The Enterprise Providers . . . . .</b>	<b>47</b>
	Introduction . . . . .	47
	Installing and Configuring an Oracle Database on Windows . . . . .	48
	Get the Installer . . . . .	48
	Remove an Old Version . . . . .	48
	Run the Installer . . . . .	49
	Create a Database . . . . .	49
	Add a Listener . . . . .	50

	Add a Service Name . . . . .	51
	Installing and Configuring an Oracle Database on Linux . . . . .	52
	Introduction . . . . .	52
	Installation . . . . .	52
	Get the Installer . . . . .	52
	Get the Installation Instructions . . . . .	53
	Check the Hardware Requirements . . . . .	53
	Check the Software Requirements . . . . .	53
	Create Required UNIX Groups and User . . . . .	54
	Create the Required Directories . . . . .	54
	Configure Kernel Parameters . . . . .	55
	Set Shell Limits For The Oracle User . . . . .	55
	Configure the Oracle Account . . . . .	56
	Run the Installer . . . . .	57
	Post-Installation Tasks . . . . .	58
	Patching . . . . .	58
	Configuration . . . . .	60
	Set the Environment . . . . .	60
	Create a Database . . . . .	60
	Configure the Database . . . . .	60
	Create a Listener . . . . .	61
	Add a Service Name . . . . .	61
	Automatic Startup of Oracle Database and Listener at System Boot Time . . . . .	61
	Installing and Configuring an Oracle Client on Linux . . . . .	62
	Introduction . . . . .	62
	Installation . . . . .	63
	Get the Installation Instructions . . . . .	63
	Get the Installer . . . . .	63
	Preliminary Steps . . . . .	64
	Run the Installer . . . . .	65
	Configuration . . . . .	65
	Set Environment Variables . . . . .	65
	Add a Local Name Service . . . . .	66
	Testing the Connection . . . . .	66
	Installing and Configuring SQL Server 2000 . . . . .	67
	Installation . . . . .	67
	Configuration . . . . .	68
	Connection . . . . .	68
	Installing and Configuring SQL Server 2005 Developer Edition . . . . .	69
	Installation . . . . .	69
	Connection . . . . .	70
<b>Chapter 4</b>	<b>The Open Source Providers . . . . .</b>	<b>71</b>
	System Requirements . . . . .	71
	Software . . . . .	71

Disk Space and Memory . . . . .	73
Windows . . . . .	73
Linux . . . . .	73
Feature Matrix . . . . .	74
Connection API . . . . .	76
Create a Username and Password for the ArcSDE Provider	
Connected to Oracle . . . . .	81
Create a Username and Password for the ArcSDE Provider	
Connected to SQL Server . . . . .	81
Connecting to the ODBC Provider . . . . .	82
Overview of the Connection Properties . . . . .	82
Create an ODBC DSN for an MS Access File . . . . .	83
Create an ODBC DSN for an Excel File . . . . .	84
Create an ODBC DSN for an Oracle Database . . . . .	85
Create an ODBC DSN for a MySQL Database . . . . .	86
Create an ODBC DSN for a SQL Server Database . . . . .	87
Command Capabilities . . . . .	88
User Management . . . . .	91
MySQL . . . . .	91
Data Store Management API . . . . .	94
Spatial Context API . . . . .	94
Exception Messages . . . . .	97
Connection Not Established . . . . .	97
Incorrect usage of DB GRANT and GLOBAL PRIVILEGES . . . . .	97

## Chapter 5 Installing RDBMS Clients Used By The Open Source

<b>Providers . . . . .</b>	<b>99</b>
Introduction . . . . .	99
Installing and Configuring an ArcSDE Client on Windows . . . . .	100
Installation . . . . .	100
Configuration . . . . .	100
Installing and Configuring an ArcSDE Client on Linux . . . . .	101
Installation . . . . .	101
Configuration . . . . .	102
Installing and Configuring the MySQL Client on Windows . . . . .	102
Installation . . . . .	102
Configuration . . . . .	102
Installing and Configuring the MySQL on Linux . . . . .	104
Installation . . . . .	104
Configuration . . . . .	104
Installing and Configuring ODBC on Linux . . . . .	105
Installation of the Driver Manager . . . . .	105
<b>Index . . . . .</b>	<b>107</b>

# The Essential FDO



You can use this document to get up and running with the Feature Data Objects (FDO) API. The Objectives section provides more information on the topics covered in this document. Comments on this document can be sent to [isd.docs@autodesk.com](mailto:isd.docs@autodesk.com).

## What Is the Feature Data Objects API?

The FDO API provides a generic interface to a number of back-end data source technologies for storing, retrieving, and updating GIS (Geographic Information System) data.

FDO provides a model for extending this interface to additional data source technologies. The generic API is extensible, and it is possible to add custom commands to a particular provider. A provider is a specific implementation of the FDO API that provides access to data stored in a particular data source technology. For example, the MySQL provider provides access to GIS data stored in an MySQL database, and the OSGeo FDO Provider for SHP provides access to GIS data stored in a SHP file. The extent to which a provider implements the FDO API is limited by the native capabilities of the underlying data source technology.

A client application creates a connection to a provider and then uses the connection object to create command objects for FDO actions such as Select. The client uses the command object to set the command parameters with the option of leaving some parameters with default values. Where appropriate such as in the case of Select, execution of the command returns a reader object containing the results of the command.

A client application can use the FDO Capabilities API to determine what services a particular provider offers. For example, the OSGeo FDO Provider for SDF supports the insertion of data, and the OSGeo FDO Provider for WMS does not.

The storage of FDO feature data can be modeled as a containment hierarchy:

- A data source may contain one or more data stores.
- A data store may contain one or more feature schema and one or more spatial contexts.
- A feature schema may contain one or more feature classes.
- A feature class contains a feature geometry property, optionally one or more non-feature geometry properties, and optionally other types of properties.

A geometry is represented using geometric constructs either defined as lists of one or more XY or XYZ points or defined parametrically, for example, as a circular arc. While geometry typically is two- or three-dimensional, it may also contain the measurement dimension (M) to provide the basis for dynamic segments. FDO Geometry is based on the *OpenGIS Simple Features Implementation Specification for SQL* (version 1.1, document 99-049), but is extended to include z and measure dimensions and circular arcs.

A schema is a logical description of the data types used to model real-world objects. A schema is not the actual data instances (that is, not a particular road or land parcel), rather it is metadata. A schema is a model of the types of data that would be found in a data store. FDO feature schemas can be written to an XML file. The FDO XML format for schema is based on the *OpenGIS Consortium Geography Markup Language* (version 2.1.1, document 02-009). The schema model is object-based using classes and properties and is independent of any physical storage format.

The API is available in two formats, enterprise and open source, on two platforms, Windows and Linux. The enterprise version of the API contains all of the open source components plus some proprietary components. The enterprise version of this document describes both open source and proprietary components. The open source version of this document describes only the open source components.

The enterprise edition is released as part of the Autodesk Map 3D and Autodesk MapGuide Enterprise products. The FDO API is not exposed in these products.

The open source edition is released as source code which may be downloaded from the Open Source Geospatial Foundation website (<http://www.osgeo.org>) in the form of gzipped tarfiles or directly from Subversion repositories. The project name is Fdo. The project documentation includes a document about how to build and install the Fdo software from source on both platforms.



# Objectives

The main objective of this document is to provide information to help you install the FDO software and then use it to build applications that connect to and configure providers. “Configure providers” means creating and deleting user accounts, granting and revoking database privileges to users, creating and destroying data stores, and creating and destroying spatial contexts. Not every provider supports configuration. See [Command Capabilities](#) (page 18).

The RDBMS-based providers with the exception of SQL Server require rdbms client software to be installed and configured in order to make the provider fully functional. This document describes the installation and configuration of these clients. Having a SQL Server client on your machine can be useful so the installation of this client is described as well.

This document identifies the classes and methods you use for connecting to and configuring providers but does not provide details about their use. For more information, see the FDO API reference documentation. Doxygen is used to generate the API reference documentation from comments wrapped in doxygen tags and embedded in the C++ headers.

The *FDO Developer's Guide* (FDG\_FDODevGuide.pdf) describes additional FDO API customization and development features for FDO applications. The developer's guide expands on the FDO API introduced in this document, further explains the role of a feature provider, and provides detailed information and examples about how to code your application.

## Related Documentation

The reference documentation includes this document as well as the following. The ReadMe and the API reference documents are formatted as HTML or CHM files on Windows and HTML on Linux. This document, the developer's guide and the open source build document are formatted as HTML and PDF.

- *FDO ReadMe*
- *FDO Developer's Guide*
- *Building FDO Open Source*
- *FDO API Reference*
- *FDO API Reference - Managed*

- *OSGeo FDO Provider for ArcSDE API Reference*
- *OSGeo FDO Provider for MySQL API Reference*
- *OSGeo FDO Provider for ODBC API Reference*
- *OSGeo FDO Provider for SDF API Reference*
- *OSGeo FDO Provider for SHP API Reference*
- *OSGeo FDO Provider for WMS API Reference*
- *Autodesk FDO Provider for Oracle API Reference* (only included with Autodesk software)
- *Autodesk FDO Provider for Raster API Reference* (only included with Autodesk software)
- *Autodesk FDO Provider for SQL Server API Reference* (only included with Autodesk software)

---

**NOTE** The FDO Provider for WFS has no WFS-specific public API; all of its functionality is accessed through the FDO API.

---

## Provider Overview

The following RDBMS-based and file-based FDO providers are included. The full provider name is listed here; subsequent references in this document may use a shortened name. Autodesk® providers are only included with commercial Autodesk software. Open source providers are also part of the Open Source Geospatial Foundation (OSGeo).

Provider Name	Description
OSGeo FDO Provider for ArcSDE	Read/write access to feature data in an ESRI ArcSDE-based data store (that is, with an underlying Oracle or SQL Server database). Supports describing schema, and inserting, selecting, updating, and deleting feature data in existing schemas; does not support creating or deleting schemas.
OSGeo FDO Provider for MySQL	Read/write access to feature data in a MySQL-based data store. Supports spatial data types and spatial query operations. Custom API can gather information, transmit exceptions, list data stores, and create connection objects. MySQL architecture supports various storage engines, characteristics, and capabilities.
OSGeo FDO Provider for ODBC	Read/write to feature data in a ODBC-based data store. Supports XYZ feature objects and can define feature classes for any relational database table with X, Y, and optionally, Z columns; does not support creating or deleting schema. Object locations are stored in separate properties in the object definition.
OSGeo FDO Provider for SDF	Read-write access to feature data in an SDF-based data store. Autodesk's geospatial file format, SDF, supports multiple features/attributes, provides high performance for large data sets, and interoperability with other Autodesk products. Supports spatial indexing and is a valid alternative to database storage. Note that this release of the SDF provider supports version 3.0 of the SDF file format.
OSGeo FDO Provider for SHP	Read/write access to existing spatial and attribute data in an ESRI SHP-based data store, which consists of separate shape files for geometry, index, and attributes. Each SHP file and its associated DBF file is treated as a feature class with a single geometry property. This is a valid alternative to database storage but does not support locking.

Provider Name	Description
OSGeo FDO Provider for WFS	Read-only access to feature data in an OGC WFS-based data store. Supports a client/server environment and retrieves geospatial data encoded in GML from one or more Web Feature Services sites. Client/server communication is encoded in XML with the exception of feature geometries, which are encoded in GML. Note that there is no public API documentation for this provider; all WFS functionality is accessible through the base FDO API.
OSGeo Provider for WMS	Read-only to feature data in an OGC WMS-based data store. Web Map Service (WMS) produces maps of spatially referenced data dynamically from geographic information, which are generally rendered in PNG, GIF, or JPEG, or as vector-based Scalable Vector Graphics (SVG) or Web Computer Graphics Metafile (WebCGM) formats.
Autodesk FDO Provider for Oracle	Read/write access to feature data in an Oracle-based data store. Supports spatial indexing, long transactions, and persistent locking. Custom API can gather provider information, transmit client services exceptions, list data stores, and create connection objects.
Autodesk FDO Provider for Raster	Read-only access to feature data in a raster-based file format. Supports various image and GIS data formats (for example, JPEG, PNG, MrSID, and others). Supports georeferenced file-based raster images and file-based grid coverages. Pixel-based images, such as satellite images, are useful underneath vector data.
Autodesk FDO Provider for SQL Server	Read/write access to feature data in a Microsoft SQL Server-based data store. A custom API supports schema read/write access and geospatial and non-geospatial data read/write access.

# Feature Description

Each FDO API feature is described in the table below.

**NOTE** This section describes the features referenced in the Feature Matrix sections in the Provider chapters.

Feature	Description
Commands	
Connection	Connect to a data store.
Capabilities	Obtain information programmatically about the features that a provider supports and does not support.
Datastore	Create, destroy, and list data stores.
Describe Schema	Get a feature schema from a data store.
Apply Schema	Create a feature schema in a data store.
Spatial Contexts	Manage the properties required for the storage and manipulation of geometric data.
Select	Use the FDO filter and expression language to select a set of features from a data store.
Select Aggregates	Selects groups of features from the data store and applies filters to each of the groups.
Insert	Create an instantiation of a feature schema class and add it to the data store.
Update	Modify a feature in the data store.

Feature	Description
Delete	Delete a feature from the data store.
SQL Command	Supports the execution of a SQL statement against an underlying RDBMS. Two execute methods are provided to distinguish between statements that return table data versus those that execute non query type operations.
Transactions	Supports transaction processing (begin, commit, rollback).
Locking	Persistent and transaction locks are supported. A persistent lock is applied, if a user requires exclusive access to an object for an extended period of time (for example, for multiple sessions). In contrast, a transaction lock is applied if exclusive access to an object is required for the duration of a transaction only. Such a transaction could cover the time required to change an object.
Long Transactions	Group conditional changes to one or more objects. A root long transaction represents permanent data. A descendent of a root long transaction represents revisions to objects where the revisions do not involve permanent changes to those objects.
<b>Schema Capabilities</b>	
Schema Overrides	This allows FDO applications to customize the mappings between Feature (logical) Schemas and the Physical Schema of the Provider data store. Schema Mappings will be customized through Schema Override sets. Each Feature Schema may have a number of associated sets, one per FDO Provider that the Feature Schema might be applied against.
Foreign Schemas	Access data stored in schemas created by third-party applications.
Associations	Describes a relationship between classes. The association has properties that describe the linkage between the classes being associated. The linkage can be one-to-one or many-to-many.
Data Property Constraints	Specify that a property or list of properties taken together have unique values for all objects of that class in a data store. A unique constraint that is based on a list of two or more properties is called a composite unique

Feature	Description
	constraint. This constraint type applies to all data property types except for Boolean, BLOB, and CLOB. A Boolean property can be included if it is part of a composite unique constraint.
Inheritance	Supports feature class hierarchies.
Object Properties	Supports class properties whose type is a class.
<b>Setup and Tools</b>	
Feature Schema Serializa- tion	Read and write Feature Schema, Schema Overrides and Spatial Context information using an OGC GML format.
Feature Data Serialization	Read and write feature data using an OGC GML format.
XML Configuration	Supports provider-specific configuration.

## System Requirements

### Operating System and RDBMS Client Software

These requirements are specified in the chapters describing the enterprise and open source providers.

### Development Environment Software

The FDO API is written in C++. The class references in this document are to C++ classes. An equivalent managed class interface is also available in Windows.

## Windows XP, SP2

Both the enterprise and open source editions of the FDO API require the use of Microsoft Visual Studio 2005 for development (Express, Standard, or Professional editions).

## Red Hat Enterprise Linux 3

The programming environment is GNU.

# Getting Started

## Install FDO

### Open Source Edition

The installation of the FDO open source edition is described in the *Building FDO Open Source* document. This document is available at <http://fdo.osgeo.org/documentation.html>. This involves getting the source, building the binaries, and then installing them.

### Enterprise Edition

FDO is installed as part of the Autodesk Map 3D and Autodesk MapGuide Enterprise installs. The FDO C++ and C# APIs are exposed in AutoCAD Map3D 2008. The FDO C# API is exposed in MapGuide Enterprise 2007 Server.



# Setup the C++ Application Development Environment

## Windows

Check out the settings in the solution (.sln) and project (.vcproj) files found in the FDO source folders.

Use the installation path to set up the environment so that the headers and libraries can be located at compile time, and the shared objects can be located at runtime.

Use the source path to set up the environment so that the debugger can locate program database files (symbol tables) and source files at runtime.

You can make the various files visible by either setting solution or project properties or by setting Visual Studio tool options or by setting an environment variable (PATH).

The first step for the properties method is to select the solution or project in the Solution Explorer, right-click to its pop-up menu and select Properties. The result is that the properties dialog is displayed. If the project properties dialog is displayed, the second step is to expand 'Configuration Properties' in the properties dialog contents pane.

The first step for the tool options method is to select 'Options...' from the Tools menu in the main menubar. The result is that the options dialog is displayed.

## Headers

In the properties dialog contents pane expand 'Configuration Properties' and then 'C/C++'. Select 'General' and edit 'Additional Include Directories'.

In the options dialog contents pane expand 'Projects and Solutions' and then 'VC++ Directories'. Select 'Include files' from the spinbox labeled 'Show directories for:'. Follow the procedure for adding a directory.

## **Libraries**

In the project properties dialog contents pane expand 'Configuration Properties' and then 'Linker'. Select 'General' and edit 'Additional LibraryDirectories'.

In the options dialog contents pane expand 'Projects and Solutions' and then 'VC++ Directories'. Select 'Library files' from the spinbox labeled 'Show directories for:'. Follow the procedure for adding a directory.

## **Dynamically Linked Libraries**

In the options dialog contents pane expand 'Projects and Solutions' and then 'VC++ Directories'. Select 'Executables' from the spinbox labeled 'Show directories for:'. Follow the procedure for adding a directory.

Use XP to set the PATH environment variable.

## **Symbol Table Files (Program Database Files)**

In the options dialog contents pane expand 'Debugging' and then 'Symbols'. Follow the procedure for adding a directory.

## **Source**

In the solution properties dialog contents pane expand 'Common Properties'. Select 'Debug Source Files'. Follow the procedure for adding a directory.

In the options dialog contents pane expand 'Projects and Solutions' and then 'VC++ Directories'. Select 'Source files' from the spinbox labeled 'Show directories for:'. Follow the procedure for adding a directory.

## **Linux**

Set LD\_LIBRARY\_PATH environment variable.

## Access the C++ API Reference Documentation

The API reference documentation has been created by using doxygen to process comments in the header files. When doxygen generates the html files, it distributes them across a generated file structure to allow for quicker access. This prevents direct linking to the class html files.

When you are asked to read FDO class reference documentation, load the FDO C++ API Reference home page, click the Classes tab in the right-hand pane and click on the class name. The home page is located in the *<install directory>/Docs/HTML/FDO\_API* directory.

## Write the Code to Connect to a Provider

Do the following:

- 1 Create a provider registry
- 2 Create a connection manager
- 3 Create a connection
- 4 Determine the provider data store type.
- 5 Get the list of dependent files
- 6 Get the connection state
- 7 Get the connection properties
- 8 Get values for the connection properties
- 9 Set the connection properties
- 10 Open a connection
- 11 Open a pending connection
- 12 Get the list of dependent files, if applicable.

## Create a Provider Registry

Read about the `GetProviders` method in the `FdoProviderRegistry` class reference. This method returns the contents of the `providers.xml` file found

in the `<install>/bin` folder on Windows and in the `/usr/local/fdo-<version>/lib` directory on Linux.

When you create the connection, you will use one of the strings returned by this method as an argument.

## Create a Connection Manager

The connection manager creates connections.

Read about the `GetConnectionManager` method in the `FdoFeatureAccessManager` class reference.

## Create a Connection

Read about the `CreateConnection` method in the `FdoConnectionManager` class reference.

## Get the Provider Data Store Type

A data store can reside in a file or database or on a website. Providers can be typed according to the location of the data store that they access. The SDE, SHP, and Raster Providers access files, and their data store type is `File`. The Oracle, MySQL, ArcSDE, and SQL Server Providers access databases, and their data store type is `DatabaseServer`. The WMS and WFS Providers access websites, and their data store type is `WebServer`. The ODBC Provider can access either a file or a database, and its data store type is `Unknown`.

You can determine the data store type of a provider after you have created a connection to it. Read about the `GetProviderDatastoreType` method in the `FdoIConnectionInfo` class reference.

## Get the List of Dependent Files

Once you have opened a connection to provider whose data store type is `File`, you can get the list of dependent files. For a connection to an sdf file, the list consists of the sdf file. For a connection to a shp file or a directory containing the shp file, the list can consist of the shp file plus other associated files of

type .dbf, .prj, and .shx. For a connection to a raster file, for example, a.jpg file, the list can consist of the jpg file plus other associated files of type .jgw and .xml.

Read about the `GetDependentFileNames` method in the `FdoIConnectionInfo` class reference.

## Get the Connection Properties

First you get the connection information. Read about the `GetConnectionInfo` method in the `FdoIConnection` class reference.

Then you get the connection properties. Read about the `GetConnectionProperties` method in the `FdoIConnectionInfo` class reference. This method returns a connection property dictionary.

Read about the methods used to access the information in the connection properties dictionary in the `FdoIPropertyDictionary` class reference. Using these methods you can find out the names of the properties, whether they are optional or mandatory, whether they have default values, their type, whether they are enumerable and, if enumerable, what the set of enumerated values is.

## Get Values for the Connection Properties

The types of properties used in connections include:

- user names
- passwords
- file path names
- URLs
- symbolic host names
- service identifiers
- data store names

Each provider has its own unique set of connection properties. Read about the connection properties in the chapter about the providers.

## Set the Connection Properties

Read about the methods used to access the information in the connection properties dictionary in the `FdoIPropertyDictionary` class reference.

## Open a Connection

Read about the `Open` method in the `FdoIConnection` class reference.

Confirm that the connection is open by calling the `GetConnectionState` method.

## Open a Pending Connection

Each of the RDBMS-based providers has a connection property, which is there to contain the name of a data store. The data store property is initially not required in order to make a connection. If the property is not set, then the resulting connection, if successful, is in a pending state.

As a result of the initial connection operation, the data store property is populated with the names of the data stores in the RDBMS, and its optionality attribute is changed to a value of required.

Use the `EnumeratePropertyValues` method on the `FdoIPropertyDictionary` object to read the set of data store names and set the data store property in the dictionary object to one of those names. Then call the `Open` method on the `FdoIConnection` object again. This will result in a fully open connection.

You can connect to an ArcSDE provider using a pending connection, you will find that there is only ever one name in the set of data store names returned as a result of the initial connection.

## **Get the List of Dependent Files**

### **Next Steps**

#### **ArcSDE Provider**

You use this provider to connect to an ArcSDE server, which is, in turn, connected to an Oracle or SQL Server data source.

The next step is to determine the number and composition of feature schema in the data store so that you can execute data maintenance (insert, update, and delete) and query (select) commands.

#### **MySQL Provider**

The next steps are to create a data store and a user, and after that, create a feature schema and execute data maintenance and query commands.

#### **SDF, SHP and ODBC Providers**

The next steps are to create a feature schema and execute data maintenance and query commands.

#### **WFS and WMS Providers**

The next steps are to determine the composition of the feature schema and execute query commands.

### **Capabilities API**

You can use this API to determine what capabilities to expose or hide in the user interface depending on the current provider.

You access the Capabilities API by way of the connection object. The availability of this API is not dependent on the connection state; it is always available. This static definition represents the maximum set of capabilities available through the current provider under optimum conditions. For example, the capabilities API says that the Oracle provider supports the update command; however, the execution of an update command could throw an exception because the user has not been granted access to the table to be updated. In this example, the update capability is conditional on user privilege. The exception message provides the additional information you need to use the capability successfully.

Use the `FdoIConnection` object to obtain the capability objects whose class names are `FdoICommandCapabilities`, `FdoIConnectionCapabilities`, `FdoIExpressionCapabilities`, `FdoIFilterCapabilities`, `FdoIRasterCapabilities`, `FdoISchemaCapabilities`, and `FdoITopologyCapabilities`.

## Command Capabilities

Use the `FdoIConnection` object's `GetCommandCapabilities` method to get the list of commands and related attributes supported by the provider.

## User Management API

You can use this API to add a user to, or drop a user from, an Oracle, MySQL, or SQL Server provider. You can also use this API to grant privileges to users. To perform these operations, you must be connected to the provider as a user with administrative privileges, and the connection must be in the pending state. You perform these operations by calling methods on the `FdoUserManager` object. You create this object by calling the static `FdoUserManager::Create` method.

## User Management Capabilities

The user management capabilities are not reflected in the capabilities API. That is, they do not show up in the information returned by the `FdoIConnection` object's `GetCommandCapabilities` method. You can use the `FdoUserManager` object's `SupportsDatastoreGrants` and `SupportsWindowsAuthentication` to control the execution of some of the



object's methods. For the rest of the object's methods you must test for the name of the provider to determine whether or not you can call the method.

## Data Store Management API

You can use this API to create a data store, destroy a data store, or list data stores. Three RDBMS providers (Oracle, MySQL, and SQL Server) and one file-based provider (SDF) support the data store commands. You can use the `FdoICommandCapabilities` object to verify this.

The procedure used for creating and destroying a data store in an RDBMS-based provider is slightly different than the procedure used for these operations in the SDF provider. The key difference is that the connection to the RDBMS-based provider is in the pending state, and the connection to the SDF provider is in the closed state.

### Oracle, MySQL, and SQL Server Providers

To create or destroy a data store, do the following:

- 1 Open a connection in the pending state to the provider as a user with administrative privileges.
- 2 Use the connection object to create the `FdoICreateDataStore` or `FdoIDestroyDataStore` command object
- 3 Use the command object to get the appropriate property dictionary.
- 4 Use the property dictionary first to determine what properties must be given values and then to set those properties values.
- 5 Call the command object's `Execute()` method.
- 6 To verify the result of these operations, use the connection object to create an `FdoIListDataStores` command. Calling the `Execute` method on this object returns a data store property reader containing a list of the data stores in the data source. By default this list includes all of the data stores in the data source. Optionally, you can exclude the data stores not created using the `FdoICreateDataStore` command.

The difference between data stores created using the `FdoICreateDataStore` command and ones created by non-FDO means (for example, from the SQL command-line) is that the former contain a set of FDO-specific tables whose names are prefixed with `F_` or `f_`, and which hold FDO metadata. You can

view the list of tables created by the `FdoICreateDataStore` command using the following operations for each provider type:

- **Oracle:** `sql> select owner,table_name from dba_tables where owner = '<data store name>';`
- **MySQL:** `mysql> show tables from <database name>;`
- **SQL Server:** In the Enterprise Manager application window, under Console Root , click Microsoft SQL Servers, click SQL Server Group, click *SQL\_Servername*, click Databases , click *database\_name*, click Tables.

Depending on the provider, most of the following values are displayed:

- `f_associationdefinition`
- `f_attributedefinition`
- `f_attributedependencies`
- `f_classdefinition`
- `f_classtype`
- `f_dbopen`
- `f_lockname`
- `f_options`
- `f_sad`
- `f_schemainfo`
- `f_schemaoptions`
- `f_sequence`
- `f_spatialcontext`
- `f_spatialcontextgeom`
- `f_spatialcontextgroup`

## SDF Provider

To create or destroy a data store, do the following:

- 1 Use the `FdoICreateDataStore` command to create an SDF file and the `FdoIDestroyDataStore` command to delete it. These operations are performed with the connection in the closed state.
- 2 Verify these operations by looking in the file system. The SDF provider does not support the `FdoIListDataStores` command.

## Spatial Context API

A spatial context defines properties required for the storage and manipulation of geometric data. You use this API to manage the spatial context of a data store.

Use the connection object to create the desired spatial context command (`FdoIActivateSpatialContext`, `FdoICreateSpatialContext`, `FdoIDestroySpatialContext`, or `FdoIGetSpatialContexts`). The connection must be in the open state to execute these commands. All of the providers support the `FdoIGetSpatialContexts` command. The ArcSDE, MySQL, Oracle, and SQL Server providers support the `FdoICreateSpatialContext` and `FdoIDestroySpatialContext` commands. The SDF provider supports the `FdoICreateSpatialContext` command but not the `FdoIDestroySpatialContext` command.

## Glossary

### Tolerance

Used to determine when two points are close enough to be considered to be equivalent in certain spatial tests. In FDO, the tolerance value is used as a radius around a position rather than as a distance along an axis. In the majority of cases, the distance between the points is compared to the tolerance using the 'less than or equals to' predicate rather than the 'less than' predicate. If the X and Y tolerances are both 0.05, then point B (1.0, 1.0) is considered to be equivalent to point A (0.96, 0.96) and point C (1.04, 1.04), but is not considered to be equivalent to point D (0.96, 0.94) or point E (1.04, 1.06). The

relationship is not commutative. That is, point A is not equivalent to point C. Repeated applications of the `isEquivalentTo` predicate to the same data always yields the same result.

# The Enterprise Providers

# 2

This chapter describes how to connect to and configure the enterprise providers (Oracle, Raster, and SQL Server).

## System Requirements

### Software

The following table maps the various providers to the operating system and RDBMS versions against which they have been tested (T). The provider requires that an Oracle client is installed in order to communicate with the Oracle database. It is your choice to install just the Oracle client or the client and server combined. Connection to a remote SQL Server database is done through the standard ODBC component that is installed with the Windows operating system so that installing a SQL Server client is not a necessity. The FDO API provides user and data store management for both the Oracle and SQL Server providers. You may, nevertheless, wish to install a SQL server client.

FDO and its Providers run on the Windows Vista operating system. However, Autodesk cannot guarantee that the underlying file or database that an FDO Provider is connecting to is Vista compatible (except for the Autodesk Spatial

Data File format, SDF). Please consult the file or database vendor to verify Vista compatibility.

**WXP=Windows XP Pro; W2K=Windows 2000 Pro; W2K3=Windows 2003 Server**

Provider	WXP	W2K	W2K3	Linux	RDBMS	Notes
Oracle	T	T	T	T	Oracle 10gR2 certified; Oracle 10gR1 and Oracle 9i are NOT certified (supported but not tested). Oracle Workspace Manager (OWM) is supported up to but NOT including version 10.2.0.4.	1
SQL Server	T	T	T	n/a	SQL Server 2000 SP 4 and SQL Server 2005 are certified	
Raster	T	T	T	T	TIFF, MrSID, PNG, JPEG, JPG2K, DEM, and ECW tested.	

#### NOTE

- 1 In previous releases the use of the Oracle Workspace Manager for versioning (locking and long transactions) was enabled by default. In the current release the OWM is disabled by default and so, therefore, is support for versioning. How to reenale versioning is described in the Oracle appendix in the FDO Developers Guide.

# Disk Space and Memory

## Windows

This table shows the disk space and memory requirements for the RDBMS clients on an XP client machine.

D = Disk (MB); M = Memory (MB); N = Notes			
Component	D	M	N
Oracle 10g R1 instance ( <code>oracle.exe</code> , <code>tnslsnr.exe</code> , <code>emagent.exe.exe</code> , <code>ocssd.exe</code> )	932	259	1
Oracle 10g R1 Instant Client	94	19	1
SQL Server 2000	34		2
SQL Server 2005 Developer Edition ( <code>sqlserver.exe</code> , <code>msmdsrv.exe</code> , <code>msftesql.exe.exe</code> )	263	71	3

### NOTE

- 1 The disk size does not include the database directory. The memory size is an approximate sum of the memory sizes for an Oracle instance running as a service. The memory footprint is configurable. The processes are `oracle.exe`, `tnslsnr.exe`, and `emagent.exe`. The Oracle instance includes both client and server components. The Oracle provider uses the Oracle client to connect to a local or remote Oracle database. When the FDO application connects to Oracle, it loads the Oracle dynamically linked libraries (DLLs)
- 2 This represents the install of the client tools on an XP machine. There are no services running. The disk size for SQL Server 2000 is that of `C:\Program Files\Microsoft SQL Server\80`.
- 3 The two major components installed are database and analysis services. The database engine service is `sqlserver.exe`. The analysis service is `msmdsrv.exe`. The full-text search service is `msftesql.exe`. The disk size is that of `C:\Program Files\Microsoft SQL Server\90`, `C:\Program Files\Microsoft SQL Server\MSSQL.2`, and `C:\Program Files\Microsoft SQL Server\MSSQL.3`.

## Linux

This table shows the disk space and memory requirements for the Oracle client on an RHEL3 client machine.

**D = Disk (MB); M = Memory (MB); N = Notes**

Component	D	M	N
Oracle 10g R1 instance (ocssd, ocssd.bin, tnslnsr, ora_XXXX_<sid>)	2100	3.9	1

### NOTE

- 1 The memory value is a combination of RAM and swap. There are 14 ora\_XXXX\_<sid> processes where 'XXXX' represents alphanumeric characters and '<sid>' represents the System Identifier (SID) value given to the database during its creation.

## Feature Matrix

The main features for Commands, Schema Capabilities, and Tools and Setup of the FDO API are listed in the table. "Y" means the command or attribute is supported in that particular provider.

**NOTE** The features are described in the [Feature Description](#) (page 7) section.

Feature	Oracle	Raster	SQL Server
<b>Commands</b>			
Connection	Y	Y	Y
List Datastores	Y	Y	Y
Create Datastore	Y	Y	Y



Feature	Oracle	Raster	SQL Server
Destroy Datastore	Y	Y	Y
Describe Schema	Y	Y	Y
Apply Schema	Y	Y	Y
Destroy Schema	Y	Y	Y
Get Spatial Contexts	Y	Y	Y
Create Spatial Context	Y		Y
Destroy Spatial Context	Y		Y
Select	Y	Y	Y
Select Aggregates	Y	Y	Y
Insert	Y		Y
Update	Y		Y
Delete	Y		Y
SQL Command	Y		Y
Transactions	Y		Y
Locking	Y		Y
Long Transactions	Y		Y
Schema Capabilities			

Feature	Oracle	Raster	SQL Server
Schema Overrides	Y		Y
Foreign Schemas	Y		Y
Associations	Y		Y
Data Property Constraints	Y		Y
Inheritance	Y		Y
Object Properties	Y		Y
<b>Setup and Tools</b>			
Feature Schema Serialization	Y	Y	Y
Feature Data Serialization	Y	Y	Y
XML Configuration	Y	Y	Y

## Connection API

The following table shows the connection properties for each provider. “T” means true, and “F” means false. For important details, refer to the associated Notes.

DV=Default Value; R=Required; P=Protected; FN=FileName; FP=FilePath; D=Datastore; E=Enumerable								
Provider/Properties	DV	R	P	FN	FP	D	E	Notes
Autodesk FDO Provider for Oracle								

**DV=Default Value; R=Required; P=Protected; FN=FileName; FP=FilePath; D=Datastore; E=Enumerable**

Provider/Properties	DV	R	P	FN	FP	D	E	Notes
Username		T	F	F	F	F	F	1
Password		T	F	F	F	F	F	1
Service		T	F	F	F	F	F	2
DataStore		F	F	F	F	T	T	3
Autodesk FDO Provider for Raster								4
DefaultRasterFileLocation		F	F	F	F	F	F	5
Autodesk FDO Provider for SQL Server								6
Username		F	F	F	F	F	F	7
Password		F	T	F	F	F	F	7
Service		T	F	F	F	F	F	8
DataStore		F	F	F	F	T	T	9

---

**NOTE**

- 1 See the User Management API topic for information about adding a username and password.
- 2 The Service property value identifies an entry in the *tnsnames.ora* file located in *<Oracle Home>\NETWORK\ADMIN.* directory. An entry has a format of *<net service name> = (...)*. The entry was created by using the Net Configuration Assistant to add a local net service name.
- 3 An Oracle instance can contain many data stores. You can obtain the list of data stores by opening a pending connection to the Oracle service as described in the Getting Started topic.
- 4 The raster file formats supported by this provider are listed in [File Formats Supported by Raster Provider](#) (page 31).
- 5 This is a file name. The named file is in the local file system.
- 6 The SQL Server provider uses the Windows ODBC component, in particular, the *SQLDriverConnect* function, to connect to a SQL Server instance.
- 7 If your Windows user name has been added to the remote SQL Server by an administrator or you are connecting to the SQL Express instance on your local machine, you do not have to provide values for the Username and Password properties. If you are connecting by way of SQL Server Authentication, you must provide a user name and password that has been added by an administrator. If you connect to a SQL Server instance with administrator privileges, you can use the FDO User Management API to add users.
- 8 If you can see the server name in the list of SQL Servers known to the ODBC Data Source Administrator, then you can connect to it using FDO. Use the name of the remote SQL Server as it appears in the list as the value of the Service property. If you want to connect to the SQL Express instance running on the local host, use *<hostname>\SQLEXPRESS* or *.\SQLEXPRESS* as the value of the Service property. Be sure the the SQL Server service is running on the local machine. To see the list of remote SQL Servers, do the following:
  - 1 In the Windows Start menu, click Settings, click Control Panel, click Administrative Tools, click Data Sources (ODBC). The ODBC Data Source Administrator dialog box is displayed.

- 2 In the ODBC Data Source Administrator dialog box, click the User DSN tab, click Add. The Create New Data Source dialog box is displayed.
  - 3 In the Create New Data Source dialog box, scroll down and select SQL Server, click OK. The Create a New Data Source to SQL Server dialog box is displayed.
  - 4 In the Create a New Data Source to SQL Server dialog box, click the down arrow-head on the spin box labeled Server to make the list of SQL Servers visible.
- 
- 9 In SQL Server terms this is a database. If your connection to the SQL Server is in a pending state, you can create a data store using the [Data Store Management API](#) (page 19). Once you have created the data store, you can connect to it by setting the DataStore connection property and calling the connection object's Open method again.

---

## File Formats Supported by Raster Provider

The following file formats are supported by the Raster Provider.

Format File	Extension	Description
JPEG	.jpg, .jpeg	Joint Photographic Experts Group
JPG2K	.jp2, .j2p, .j2c, .j2k, .jpc, .jpk	Joint Photographic Experts Group
MrSID	.sid	Multi-Resolution Seamless Image Database
PNG	.png	Portable Network Graphic
TIFF	.tif, .tiff	Tagged Image File Format
DEM	.dem	Digital Elevation Model
ECW	.ecw	Enhanced Compressed Wavelet

Format File	Extension	Description
DTED	.dt0, dt1, .dt2	Digital Terrain Elevation Data (Windows Only)
ESRI Grid	.asc, .adf	ESRI ASCII and Binary Grid Files (Windows Only)
NITF	.ntf	National Imagery Transmission Format (Windows Only)

## Command Capabilities

The list of command class names is shown in the first table. For descriptions of what the commands do, see the API reference documentation. The command attributes are shown in the second table. For descriptions of what the attributes mean, see entry for the `FdoICommandCapabilities` class in the API reference documentation. “Y” means the command or attribute is supported in that particular provider.

Command Class Name	Oracle	Raster	SQL Server
<code>FdoISelect</code>	Y	Y	Y
<code>FdoISelectAggregates</code>	Y	Y	Y
<code>FdoIInsert</code>	Y		Y
<code>FdoIDelete</code>	Y		Y
<code>FdoIUpdate</code>	Y		Y
<code>FdoIDescribeSchema</code>	Y	Y	Y
<code>FdoIDescribeSchemaMapping</code>		Y	
<code>FdoIApplySchema</code>	Y		Y
<code>FdoIDestroySchema</code>	Y		Y

Command Class Name	Oracle	Raster	SQL Server
FdoIActivateSpatialContext	Y		Y
FdoICreateSpatialContext	Y		Y
FdoIDestroySpatialContext	Y		Y
FdoIGetSpatialContexts	Y	Y	Y
FdoICreateDataStore	Y		Y
FdoIDestroyDataStore	Y		Y
FdoIListDataStores	Y		Y
FdoICreateSDFFile			
FdoISQLCommand	Y		Y
FdoIAcquireLock	Y		
FdoIGetLockInfo	Y		
FdoIGetLockedObjects	Y		
FdoIGetLockOwners	Y		
FdoIReleaseLock	Y		
FdoIActivateLongTransaction	Y		
FdoICommitLongTransaction	Y		
FdoICreateLongTransaction	Y		

Command Class Name	Oracle	Raster	SQL Server
FdoIDeactivateLongTransaction	Y		
FdoIFreezeLongTransaction			
FdoIGetLongTransactions	Y		
FdoIRollbackLongTransaction	Y		
FdoICreateSpatialIndex	Y		Y
FdoIDestroySpatialIndex	Y		Y
FdoIGetSpatialIndexes	Y		Y

**NOTE** The capability to create and destroy a data store is associated with the capability to create and destroy a feature schema. To create a data store you specify values for data store properties. For more information about these properties are discussed in [Data Store Management API](#) (page 19).

Attribute	Oracle	Raster	SQL Server
Parameters			
Timeout			
SelectExpressions	Y		Y
SelectFunctions	Y		Y
SelectDistinct	Y		Y
SelectOrdering	Y		Y
SelectGrouping	Y		Y



## User Management API

You can use this API to add a user to, or drop a user from, an Oracle, MySQL, or SQL Server provider. You can also use this API to grant privileges to users. To perform these operations, you must be connected to the provider as a user with administrative privileges, and the connection must be in the pending state. You perform these operations by calling methods on the `FdoUserManager` object. You create this object by calling the static `FdoUserManager::Create` method.

## User Management SQL Commands

The following table maps methods to providers. The Operation column contains the name of methods in class `FdoUserManager`. If the method returns a boolean, the table cell contains the boolean value returned by the provider. If the method performs an operation and the provider supports the operation, the table cell contains the SQL statements executed by the provider.

Operation	MySQL	Oracle	SQL Server	Notes
AddUser(name, password)	create user 'name' identified by password	create user name identified by password	exec sp_addlogin @loginame = 'name', @passwd = 'pwd'; exec sp_addsrvrolemember @loginame = 'name' @rolename = 'dbcreator'	1,2,3
ApplyWindowsUser(domain, name)	Not supported	Not supported	EXEC sp_grantlogin @loginame = 'name' "domain/name"	
AssignRoleOrPrivilege(name, roleOrPriv)	grant roleOrPriv *.* to name	grant roleOrPriv to name	Not supported	4,5

Operation	MySQL	Oracle	SQL Server	Notes
DropUser(name)	drop user <i>name</i>	drop user <i>name</i>	exec sp_revoked- baccess @name_in_db = ' <i>name</i> '; EXEC sp_droplogin @lo- giname = ' <i>name</i> '	6
GetDatastores()	FdoIListData- stores	FdoIListData- stores	FdoIListData- stores	7
GetDbUsers()	select user as name from mysql.user	select grantee as name from sys.dba_role_privs order by grantee	use master and then for each user on server, drop table Temp_RoleMem- bers52; create table Temp_Role- Members52 (DbRole varchar(50), Mem- berName varchar(128), MemberSID varchar(50)); in- sert into Temp_RoleMem- bers52 execute sp_helprolemem- ber; drop table Temp_RoleMem- bers52	8
GetDbUsers(in- Datastore)	select distinct grantee as name from informa- tion_schema.schema_priv- ileges where table_schema = <i>inDatastore</i>	Not supported	use <i>inDatastore</i> ; select name from sysusers where hasdbaccess=1 or- der by name asc;	9

Operation	MySQL	Oracle	SQL Server	Notes
GetRolesOrPrivileges()	select distinct privilege_type from information_schema.user_privileges where grantee like "root%" order by privilege_type	select role as role from sys.dba_roles order by role	select name as role from sysusers where issqlrole=1 order by name	10,11, 12
GrantAccessToDatastore(user, datastore)	grant all on <i>datastore.*</i> to <i>user</i>	Not supported	USE <i>datastore</i> ; exec sp_grantdbaccess @loginame = ' <i>user</i> '; exec sp_addrolemember @rolename = 'db_datareader' @membername = ' <i>user</i> '	13
GrantAccessToDatastore(user, datastore)		.	exec sp_addrolemember @rolename = 'db_datawriter' @membername = ' <i>user</i> '; exec sp_addrolemember @rolename = 'db_ddladmin' @membername = ' <i>user</i> '	13
GrantAccessToDatastore(user, datastore, priv)	grant <i>priv</i> on <i>datastore.*</i> to <i>user</i>	Not supported	USE <i>datastore</i> ; exec sp_grantdbaccess @loginame = ' <i>user</i> '; exec sp_addrolemember, @rolename = ' <i>priv</i> ', @member-	14

Operation	MySQL	Oracle	SQL Server	Notes
			<pre>name = 'user'; use master</pre>	
RevokeAccessToDatastore(user, datastore, priv)	<pre>revoke <i>priv</i> on <i>datastore</i>.<i>*</i> from <i>user</i></pre>	Not supported	<pre>use <i>datastore</i>; exec sp_droprole- member @rolename = '<i>priv</i>', @member- name = '<i>user</i>'; use master</pre>	15
RevokeRoleOrPrivilege(user, roleOrPriv)	<pre>revoke <i>roleOrPriv</i>- <i>ilege</i> on <i>.*</i> from <i>user</i>.</pre>	<pre>revoke <i>roleOrPriv</i> from <i>user</i></pre>	Not supported	16
SupportsDatastoreGrants()	True	False	True	17
SupportsWindowAuthentication()	False	False	True	18

---

**NOTE**

- 1 A new MySQL user is created with no privileges. Assume that you have created a new user called newuser on the MySQL instance on your machine. Log in by executing `C:\>mysql -u newuser -p`. Then execute `mysql> show grants for current_user;`. The result string is "GRANT USAGE ON \*.\* TO 'newuser'@'%' IDENTIFIED BY PASSWORD '\*94BDCEBE19083CE2A1F959FD02F964C7AF4CFC29'". This means that newuser has no privileges and can only log in locally. Call `FdoUserManager::GrantAccessToDatastore()` with user name and data store arguments to give the user default privileges.
- 2 The new Oracle user is assigned the `F_USER_ROLE` role. The SQL statement used to obtain this information is `select granted_role from sys.dba_role_privs where grantee = 'username';`. A new Oracle user is assigned the 'unlimited tablespace,' 'execute any indextype,' and 'execute any procedure' privileges. The SQL statement used to obtain this information is `select privilege from sys.dba_sys_privs where grantee = 'username';`. A new Oracle user is assigned a long list of privileges as a result of being assigned the role of `F_USER_ROLE`. You can determine the contents of this list using the SQL statement `select privilege from sys.role_sys_privs where role = 'F_USER_ROLE';`. The list is drop user, alter user, analyze any, create user, alter session, drop any view, create session, drop any index, drop any table, grant any role, lock any table, alter any index, alter any table, create any view, create any index, create any table, create procedure, delete any table, drop any synonym, insert any table, select any table, update any table, drop any sequence, alter any sequence, create any synonym, create any sequence, grant any privilege, select any sequence, global query rewrite, and grant any object privilege. The `F_USER_ROLE` also has default table privileges. You can determine what these are by the SQL statement `select table_name, privilege from sys.role_tab_privs where role = 'rolename';`
- 3 A new SQL Server user created using the stored procedure `sp_login` is given one system privilege `db_creator`. The user can login using SQL Server authentication. Call `GrantAccessToDatastore(username, datastore)` to grant the user default privileges for the data store.
- 4 To confirm the results of this method call, call `GetDBUsers()`.
- 5 User privileges extend across all data stores defined in the Oracle instance.

- 6 To drop a windows user (SQL Server only), format the argument as `L"domain name\user name"`
- 7 You can get the same information from the connection properties dictionary for the DataStore property. The MySQL SQL command `mysql> show databases;` lists the database (data store) names. In Oracle, user names and data store names are classed as Oracle users. The Oracle SQL command `SQL> select username from dba_users;` shows data store and user names
- 8 All users are listed. SQL Server reports the user privileges as well as the user name. If you are connected to MySQL and you created the user with `AddUser`, `USAGE` is shown as the privilege. If you are connected to SQL Server and you created the user with `AddUser`, `db_creator` is shown as the privilege.
- 9 All users in a particular data store are listed along with the privileges that they have in the data store. If connected to MySQL and you used the two-argument version of `GrantAccessToDatastore`, the privileges are listed in the note for that method.
- 10 The list of supported privileges for MySQL consists of `ALTER`, `ALTER ROUTINE`, `CREATE`, `CREATE ROUTINE`, `CREATE TEMPORARY TABLES`, `CREATE USER`, `CREATE VIEW`, `DELETE`, `DROP`, `EXECUTE`, `FILE`, `INDEX`, `INSERT`, `LOCK TABLES`, `PROCESS`, `REFERENCES`, `RELOAD`, `REPLICATION CLIENT`, `REPLICATION SLAVE`, `SELECT`, `SHOW DATABASES`, `SHOW VIEW`, `SHUTDOWN`, `SUPER`, and `UPDATE`.
- 11 The list of supported roles for Oracle consists of `AQ_ADMINISTRATIVE_ROLE`, `AQ_USER_ROLE`, `AUTHENTICATED_USER`, `CONNECT`, `CTXAPP`, `DBA`, `DELETE_CATALOG_ROLE`, `DMUSER_ROLE`, `DM_CATALOG_ROLE`, `EJBCLIENT`, `EXECUTE_CATALOG_ROLE`, `EXP_FULL_DATABASE`, `F_SCHEMA_ROLE`, `F_USER_ROLE`, `GATHER_SYSTEM_STATISTICS`, `GLOBAL_AQ_USER_ROLE`, `HS_ADMIN_ROLE`, `IMP_FULL_DATABASE`, `JAVADEBUGPRIV`, `JAVAIDPRIV`, `JAVASYSPRIV`, `JAVAUSERPRIV`, `JAVA_ADMIN`, `JAVA_DEPLOY`, `LOGSTDBY_ADMINISTRATOR`, `MGMT_USER`, `OEM_MONITOR`, `RECOVERY_CATALOG_OWNER`, `RESOURCE`, `SCHEDULER_ADMIN`, `SELECT_CATALOG_ROLE`, `WKUSER`, `WM_ADMIN_ROLE`, and `XDBADMIN`.
- 12 The list of supported privileges for SQL Server consists of `db_accessadmin`, `db_backupoperator`, `db_datareader`, `db_datawriter`, `db_ddladmin`, `db_denydatareader`, `db_denydatawriter`, `db_owner`, `db_securityadmin`, and `public`.

- 13 This overloaded method takes two arguments: a user name and a data store name. It grants a set of default privileges to the user when connected to that data store. For MySQL the privileges are alter, alter routine, create, create routine, create temporary tables, create view, delete, drop, execute, index, insert, lock tables, references, select, show view, and update. For SQL Server the privileges are db\_datareader, db\_datawriter, and db\_ddladmin. Call `GetDBUsers(<datastore>)` to verify the result.
- 14 This overloaded method takes three arguments: a user name, a data store name, and a privilege name. It grants the specified privilege to that user for the named data store. The MySQL provider does not allow you to grant any privilege other than those assigned by a call to `GrantAccessToDatastore(<user>, <datastore>)`. The SQL Server provider does allow you to grant any of the supported privileges with the exception of the public privilege. Call `GetDBUsers(<datastore>)` to verify the result.
- 15 This method takes three arguments: a user name, a data store name, and a privilege name. It revokes the specified privilege from that user for the named data store. If the role is set to "\*", the provider revokes all privileges in the data store from the user. In this case the SQL Server provider also drops the database user from the data store; the user's login status is not changed.
- 16 This method takes two arguments: a user name and a role or privilege name. It revokes the specified role or privilege.
- 17 If True, you can call the two `GrantAccessToDatastore` methods and the `RevokeAccessToDatastore` method.
- 18 If True, you can call the `ApplyWindowUser` method.

## Data Store Management API

This table shows the data store management properties for each provider. "T" means true, and "F" means false.

DV=DefaultValue; R=Required; P=Protected; FN=FileName; D=Datastore; E=Enumerable

Provider/Properties	DV	R	P	FN	D	E	Notes
---------------------	----	---	---	----	---	---	-------

Autodesk.Oracle.3.0							
---------------------	--	--	--	--	--	--	--

DV=DefaultValue; R=Required; P=Protected; FN=FileName; D=Datastore; E=Enumerable							
Provider/Properties	DV	R	P	FN	D	E	Notes
DataStore		T	F	F	T	F	1
Password		T	T	F	F	F	2
Description		F	F	F	F	F	
LtMode	OW	F	F	F	F	T	3
TableSpace		F	F	F	F	F	4
Autodesk.SqlServer.3.0							
DataStore		T	F	F	T	F	1
Password		T	T	F	F	F	2
Description		F	F	F	F	F	
LtMode	FDO	F	F	F	F	T	3
LockMode	FDO	F	F	F	F	T	5



---

**NOTE**

- 1 A data store is a container in the data source to which you can apply a feature schema.
  - 2 A data store is password protected.
  - 3 LtMode stands for Long Transaction Mode. .
    - For Oracle the possible values are OWM and NONE. OWM stands for Oracle Workspace Manager. NONE means that long transactions and locking are not supported; OWM means that they are.
  - 4 If no value is given to the `TableSpace` property, the data store is created in the default `USERS` tablespace. Otherwise, the data store is created in the named tablespace.
  - 5 LockMode indicates whether persistent locks are supported. FDO means that the class tables have additional columns for maintaining persistent locks. NONE means that persistent locks are not supported. As with Oracle, LtMode pertains to long transactions. LockMode and LtMode must both be either FDO or NONE. That is, they are either both enabled or both disabled.
- 

## Spatial Context API

The following table defines default values for the spatial context properties for the providers (Oracle and SQL Server) that support the creation of a data store and an example set of spatial context properties extracted from an image file by the Raster provider.

Properties/Provider	Oracle	SQL Server	Raster	Notes
Coordinate system	Cartesian	Cartesian	Cartesian	
Extent type	Static	Static	Static	1
Extent	Not NULL	Not NULL	Not NULL	2
Unit dimension	XY	XY	XY	3

Properties/Provider	Oracle	SQL Server	Raster	Notes
XY tolerance	.001	.001	.001	4
Z tolerance	.001	.001	.001	4
Active	False	False	T	5

#### NOTE

- 1 A static extent type means that the extent is defined at the time of the creation of the spatial context using the `FdoICreateSpatialContext` command. A dynamic extent type means that the extent may change as features are added so that it always encloses all of the features.
- 2 The default extents expressed in AGF text format for the various providers are as follows:
  - Oracle and SQL Server: POLYGON ((-2000000 -2000000, 2000000 -2000000, 2000000 2000000, -2000000 2000000, -2000000 -2000000))
  - Raster: POLYGON ((-10000000 -10000000, 10000000 -10000000, 10000000 10000000, -10000000 10000000, -10000000 -10000000))
- 3 The dimensionality is that of the extent. It is not set directly.
- 4 See [Tolerance](#) (page 21).
- 5 The Oracle and SQL Server providers can have multiple spatial contexts defined and can, therefore, show contexts in the reader that are not active.

## Exception Messages

### Connection Not Established

This exception message occurs when there is a closed connection to either SQL Server and a call is made to `EnumeratePropertyValues` on the connection properties dictionary object for the `DataStore` property.

## Membership of the public role cannot be changed

This exception message occurs when there is a pending connection to SQL Server and a call is made to `GrantAccessToDatastore(<user>, <datastore>, L"public")`.



# **Installing RDBMS Clients Used By The Enterprise Providers**

# **3**

## **Introduction**

The operation of the Oracle provider requires the installation of an Oracle database client on the same machine. This chapter describes the installation of Oracle 10g release 1 databases on Windows XP and Linux x86 (RHEL3) and the installation of an Oracle 10g full client on Linux x86 (RHEL3).

The operation of the SQL Server provider does not require the installation of a SQL Server client. This provider uses ODBC drivers installed with Windows XP to talk to the SQL Server database. You will find the availability of a SQL Server client on your machine useful so the installations of a SQL Server 2000 client and a SQL Server 2005 client are described.

# Installing and Configuring an Oracle Database on Windows

## Get the Installer

Use the following information to get the setup binary used to install Oracle 10g release 1. You can install the full database, the client, or the instant client. The following steps describe how to install the full database.

- 1 Go to  
*<http://www.oracle.com/technology/software/products/database/oracle10g/index.html>*
- 2 Click Oracle Database 10g Release 1 (10.1.0.2) for Microsoft Windows (32-bit), which displays the “Oracle Technology Network Developer License Terms” page.
- 3 Read the export restriction statements and the license agreement. Click the checkbox and click I Accept, which displays the “Oracle Database 10g Release 1 (10.1.0.2)” page.
- 4 Click the 10g\_win32\_db.zip link, which displays the File Download dialog box asking you if you want to Open or Save this file. Click Save, which displays a file system browser dialog box where you can save the file to the desktop.
- 5 Unzip the file to the desktop. This action results in a folder called Disk1 being created on the desktop. This folder contains setup.exe.

## Remove an Old Version

Before installing the new version of Oracle, you must remove the old one. The following procedure deinstalls the entire Oracle database.

- 1 Select Universal Installer from the menu path Start ► Programs ► Oracle - OraDb10g\_home1 ► Oracle Installation Products.
- 2 In the installer window click Deinstall Products. Make the appropriate selections in the dialog boxes.
- 3 In the installer window click Cancel.

- 4 Delete or remove the Oracle folder from the file system. This folder will be recreated when you install the new version of Oracle. You may encounter a permission problem when you attempt to do this. In one case the problem was that an Oracle DLL was still in use; a restart fixed this problem.

## Run the Installer

Double-click Disk1 ► setup.exe. The installer creates, by default, a database called orcl. Uncheck this option and use the Database Configuration Assistant later to create a database. As a result of the install the OracleCSService is started.

## Create a Database

Select Database Configuration Assistant from the menu path

Start ► Programs ► Oracle - OraDb10g\_home1 ► Configuration and Migration Tools.

In step 3 Database Identification when you type a fully qualified name (<root>.domainname) in the text box labeled Global Database Name, you will see the <root> of that fully qualified name appear in the text box labeled SID (Oracle System Identifier). The fully qualified name used for this example is 'r10102.ads.autodesk.com'.

---

**NOTE** Keep the root of the fully qualified name to 8 characters or less. Oracle will truncate the root name for some purposes and not others. Save yourself some confusion.

---

In the step that creates the system accounts, use the default accounts (SYS, SYSTEM, DBSNMP, and SYSMAN). You can create your own accounts later using the Fdo User Manager API. Be sure to specify a password for the default accounts. This example uses 'test' for the password.

Verify that the database and database console services are running.

Click on the Services icon in the *Administrative Tools* folder. The access path is Start ► Programs ► Settings ► Control Panel ► Administrative Tools. In the Services window, you will see two services whose names are something like OracleServiceR10102 and OracleDBConsoleR10102. Each should have a status of Started.

## Add a Listener

Use the Net Configuration Assistant to add a service which listens on a TCP port for database connect requests.

Select Net Configuration Assistant from the menu path

Start ► Programs ► Oracle - OraDb10g\_home1 ► Configuration and Migration Tools.

Use this assistant to add a Listener Configuration. Use the default string 'LISTENER' for the Listener name value. Select TCP as the protocol. Use the default port 1521. Use the symbolic host name for your machine for the HOST value.

Verify that the listener service is running

Click on the Services icon in the *Administrative Tools* folder. The access path is Start ► Programs ► Settings ► Control Panel ► Administrative Tools. In the Services window, you will see a service whose name is something like OracleOraDb10g\_home1TNSListener and whose status is Started.

Edit the *listener.ora* file in *C:\oracle\product\10.1.0\Db\_1\NETWORK\ADMIN*. You will find an entry for LISTENER that will look like the following.

```
LISTENER =  
  
  (DESCRIPTION_LIST =  
  
    (DESCRIPTION =  
  
      (ADDRESS_LIST =  
  
        (ADDRESS = (PROTOCOL = TCP) (HOST = otwshak.ads.autodesk.com) (PORT  
          = 1521))  
  
      )  
  
      (ADDRESS_LIST =  
  
        (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC))  
  
      )  
  
    )  
  
  )
```



## Add a Service Name

Use the Net Configuration Assistant to add a local service name. This name represents a database running on a host. You will use this name in two ways. The first way is as the value for the Service property when connecting to the FDO Provider for Oracle. The second way is in a sqlplus connect string.

This procedure can be used to configure a service name for a local database or for a remote one. This example configures a service name for the local database whose installation is described above.

Select Net Configuration Assistant from the menu path

Start ► Programs ► Oracle - OraDb10g\_home1 ► Configuration and Migration Tools.

Use the assistant to add a Local Net Service Name Configuration. The assistant recommends that you use the Global Database Name as the service name. You will recall that we assigned 'r10102.ads.autodesk.com' as the Global Database Name. Use TCP as the protocol value. Use 1521 as the port value. Use the symbolic name of your host as the host value.

Verify that the service name is configured by locating an entry for it in the *tnsnames.ora* file and by connecting to it using sqlplus.

Edit the *tnsnames.ora* file in *C:\oracle\product\10.1.0\Db\_1\NETWORK\ADMIN*. You will an entry for R10102 or whatever name that you actually used for the service. The entry will look like the following.

```
R10102 = (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PRO
TOCOL = TCP) (HOST = otwshak.ads.autodesk.com) (PORT = 1521)) )
(CONNECT_DATA = (SERVICE_NAME = r10102.ads.autodesk.com)
) )
```

To verify that you can connect to the service, execute the following commands in a cmd.exe window.

- 1 sqlplus/nolog
- 2 sqlplus>connect sys/test@r10102 as sysdba

Occasionally you may get an ORA-12638 error whose message is Credential retrieval failed. A system restart should fix this.

You are now ready to try connecting to the service using the Fdo Oracle provider.

# Installing and Configuring an Oracle Database on Linux

## Introduction

What follows is a detailed description of the installation and configuration of an Oracle database on a RHEL3 machine.

## Installation

### Get the Installer

Use the following information to get the installer used to install Oracle 10g on Linux. You can install the full database, the client, or the instant client. The following steps describe how to get the installer used to install the full database.

To get the installer:

- 1 Go to <http://www.oracle.com/database/index.html>
- 2 Click FREE DOWNLOAD Download Oracle Database 10g Release 2
- 3 Click Oracle Database 10g Release 1 (10.1.0.3) for Linux x86
- 4 Click the checkbox labeled YES, I accept the License Terms and Export Restrictions and I acknowledge that I have reviewed and understood the agreement and agree to use the language I selected in entering into this agreement.
- 5 Click I Accept the licence agreement. This takes you to <http://www.oracle.com/technology/software/products/database/oracle10g/htdocs/linuxsoft.html>
- 6 Click ship.db.lnx32.cpio.gz . This will download the file to your Linux machine. The following two instructions taken from the web page unbundle the cpio file into a directory called *Disk1* in your home account on your Linux machine.
- 7 `linuxTerminal> gunzip ship.db.lnx32.cpio.gz`

```
8 linuxTerminal> cpio -idmv < ship.db.lnx32.cpio
```

## Get the Installation Instructions

This installation involves many manual steps. You will need the installation instructions. The following topics correspond to the sequence of instructions in the installation document.

- 1 On the same web page as the `ship.db.lnx32.cpio.gz` link, click here in Installation guides and general Oracle Database 10g documentation can be found here. This loads the <http://www.oracle.com/technology/documentation/database10g.html> page.
- 2 Scroll down to Oracle Database 10g Release 1 (10.1) Documentation for Linux x86
- 3 Download the PDF for Oracle Database Quick Installation Guide 10g Release 1 (10.1.0.3) for Linux x86

## Check the Hardware Requirements

- The `grep MemTotal /proc/meminfo` result is `MemTotal: 1027396 kB`, which is greater than the required 524288 kB.
- The `grep SwapTotal /proc/meminfo` result is `SwapTotal: 2040244 kB`, which is greater than the required 409600 kB.
- The `du -h /tmp` result is 3.9 GB, which is greater than the required 2.5 GB.
- The `du -h` result is 3.9 GB on a single filesystem, which is greater than the required 3.7 GB.

## Check the Software Requirements

The packages found on this machine satisfy the stated requirements. The relevant output from `rpm -qa` is:

- `kernel-2.4.21-40.EL`
- `gcc-3.2.3-54`

- gcc-c++-3.2.3-54
- glibc-2.3.2-95.39
- make-3.79.1-17.1
- openmotif-2.2.3-5.RHEL3.3
- setarch-1.3-1
- compat-db-4.0.14-5.1
- compat-gcc-7.3-2.96.128

---

**NOTE** The installer did complain about openmotif-2.2.3-5.RHEL3.3 but was ignored.

---

## Create Required UNIX Groups and User

The *oinstall* and *dba* groups were created. The user *oracle* was created and added to *oinstall* as the primary group and to *dba* as a secondary group.

---

**NOTE** These operations were done using the Users and Groups application rather than the command-line tools `groupadd`, `useradd`, and `usermod` because the latter returned puzzling results.

---

## Create the Required Directories

You can put the binaries and the data on separate filesystems given the right conditions. These conditions did not exist on the target machine so both were put on the same filesystem.

```
1 [root] # mkdir -p /app/oracle
2 [root] # mkdir /oradata
3 [root] # chown -R oracle:oinstall /app/oracle
4 [root] # chown -R oracle:oinstall /oradata
5 [root] # chmod -R 775 /app/oracle
6 [root] # chmod -R 775 /oradata
```

## Configure Kernel Parameters

- The `sysctl -a | grep sem` result showed the `semopm` parameter value as 32, which is less than the required value of 100. So the line `kernel.sem = 240 32000 100 128` was added to `/etc/sysctl.conf`
- The `sysctl -a | grep shm` result was initially interpreted as valid. Later the installer complained that the `shmmax` parameter value of 33554432 was less than the required 2147483648, which is supposed to be half the size of physical memory in bytes. So the line `kernel.shmmax = 2147483648` was added to `/etc/sysctl.conf`. The size of physical as reported by `/proc/meminfo` is 1027396 kB or 1052053504 bytes, half of which is 526026752. There is a discrepancy here.
- The `sysctl -a | grep file-max` result is valid.
- The `sysctl -a | grep ip_local_port_range` result showed the range to be 32768 to 61000, which is different than the required range of 1024 to 65000. So the line `net.ipv4.ip_local_port_range = 32768 65000` was added to `/etc/sysctl.conf`. The lower limit of 32768 was left to satisfy the stipulation that if any existing value is higher than the required value, the existing value is to be left unchanged.

## Set Shell Limits For The Oracle User

The following lines were added to `/etc/security/limits.conf`.

```
oracle soft nproc 2047
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
```

The following line was added to `/etc/pam.d/login`.

```
session required /lib/security/pam_limits.so
```

The following lines were added to `/etc/profile`.

```
if [ $USER = "oracle" ]; then
    if [ $SHELL = "/bin/ksh" ]; then
        ulimit -p 16384
```

```

        ulimit -n 65536
    else
        ulimit -u 16384 -n 65536
    fi
fi

```

The following lines were added to */etc/csh.login*.

```

if ( $USER == "oracle" ) then
    limit maxproc 16384
    limit descriptors 65536
endif

```

## Configure the Oracle Account

The installer was unbundled in a user's directory. It must be run from the oracle account. Move the installer directory to the oracle and change the user and group ownership to *oracle* and *oinstall*. Login as the oracle user.

- 1 [root] # chown -R oracle:oinstall Disk1
- 2 [root] # mv Disk1 /home/oracle/

The following commands were run.

- 1 The [oracle] # echo \$SHELL result is /bin/bash.
- 2 The line `umask 022` was added to *.bash\_profile*.
- 3 The new `umask` value was added to the environment: `./ .bash_profile`.
- 4 [oracle] # ORACLE\_BASE=/app/oracle
- 5 [oracle] # ORACLE\_SID=r10103
- 6 [oracle] # export ORACLE\_BASE ORACLE\_SID
- 7 [oracle] # unset ORACLE\_HOME
- 8 [oracle] # unset TNS\_ADMIN

## Run the Installer

- 1 [oracle] # cd /tmp
- 2 [oracle] # ~oracle/Disk1/runInstaller. The installer halted to announce that two prerequisite checks had failed. The installer is cancelled. The first check was for the value of the shmmax kernel parameter; this was corrected. The second check was for the presence of the openmotif-2.1.30-11 rpm; this was not corrected
- 3 [oracle] # ~oracle/Disk1/runInstaller. The installer complained again about the openmotif rpm, but was given authorization to continue. The installer halted to request that a script be run as root.
- 4 [root] # /app/oracle/product/10.1.0/Db\_2/root.sh. The following is echoed to stdout.  
Running Oracle10 root.sh script...  
The following environment variables are set as:  
ORACLE\_OWNER= oracle  
ORACLE\_HOME= /app/oracle/product/10.1.0/Db\_2  
Enter the full pathname of the local bin directory: [/usr/local/bin]:  
Copying dbhome to /usr/local/bin ...  
Copying oraenv to /usr/local/bin ...  
Copying coraenv to /usr/local/bin ...  
Creating /etc/oratab file...  
Adding entry to /etc/oratab file...  
Entries will be added to the /etc/oratab file as needed by Database Configuration Assistant when a database is created  
Finished running generic part of root.sh script. Now product-specific root actions will be performed.  
/var/opt/oracle does not exist. Creating it now.  
/etc/oracle does not exist. Creating it now.  
Successfully accumulated necessary OCR keys.  
Creating OCR keys for user 'root', privgrp 'root'..  
Operation successful.  
Oracle Cluster Registry for cluster has been initialized

```
Adding to inittab
Checking the status of Oracle init process...
Expecting the CRS daemons to be up within 600 seconds.
CSS is active on these nodes.

otwcamed
CSS is active on all nodes.

Oracle CSS service is installed and running under init(1M)
```

**5 The installer was authorized to continue, and it completed by displaying the following.**

The following J2EE Applications have been deployed and are accessible at the URLs listed below.

Ultra Search URL:

`http://otwcamed:5620/ultrasearch`

Ultra Search Administration Tool URL:

`http://otwcamed:5620/ultrasearch/admin`

iSQL\*Plus URL:

`http://otwcamed:5560/isqlplus`

iSQL\*Plus DBA URL:

`http://otwcamed:5560/isqlplus/dba`

## Post-Installation Tasks

These tasks are described in the *The Oracle Database Installation Guide*. The PDF file *B10811\_05.pdf* can be downloaded from <http://www.oracle.com/technology/documentation/database10g.html>. Chapter 4 describes the post-installation tasks.

## Patching

Patching is a required post-installation task. To get the patch follow the instructions in Chapter 4 of *The Oracle Database Installation Guide*. You will have a zip file with a name similar to *p4505133\_10105\_LINUX.zip*. Unzip the patch file to a temporary directory. The unzip operation creates a directory called *Disk1* and an html readme file. The readme contains the instructions



on how to apply the patch. Switch user to root, change the user and group ownership of the temporary directory to oracle and oinstall respectively and move the directory to the oracle home account.

Before applying the patch, backup the `$ORACLE_HOME/root.sh` script. You ran this script during the installation. The patch installer will overwrite this file.

What follows is a description of applying a patch.

- 1 Identify the existing database installation by browsing the `/etc/oratab` file. The line of interest is `*:/app/oracle/product/10.1.0/Db_2:N`. This line contains three colon-delimited fields:  
`$ORACLE_SID:$ORACLE_HOME:<N|Y>`. The comments do not specify a meaning for the asterisk in the first field. No database has been created yet. After you create a database, you will see that the asterisk has been replaced by the SID that you specified when creating it. The 'N' indicates that the `dbstart` utility will not restart the database after a system reboot.
- 2 Use the information from the `oratab` file to set the `ORACLE_SID` and `ORACLE_HOME` environment variable. In this case `ORACLE_SID` is not set since no database exists.
- 3 Shut down the Oracle databases. There are none to shut down.
- 4 Stop all listener processes. Check for listener processes. `ps -ef | grep tnslnsr`. None have been configured yet. If you had, you would stop it using `lsnrctl` utility. This utility is in the `$ORACLE_HOME/bin` directory.
- 5 Check for the Oracle Cluster Synchronization Service processes. The `ps -ef` output in this exercise shows `/bin/su -l oracle -c exec /app/oracle/product/10.1.0/Db_2/bin/ocssd` and `/app/oracle/product/10.1.0/Db_2/bin/ocssd.bin`.
- 6 Stop the two Oracle Cluster Synchronization Service processes. The `/etc/init.d/init.cssd stop`. The result is that the CRS daemon, the EVM daemon, and the CSS daemon are shut down.
- 7 Backup the databases. None exist.
- 8 Switch user to the oracle account, change to the `Disk1` created by unzipping the patch file and launch the installer: `./run_installer`.
- 9 In the Specify file locations screen the installer asks you to specify a name and a path. The path must be the value of the `ORACLE_HOME` environment variable.

- 10 The installer displays a dialog box asking you to run *root.sh* script as root and to resume installer execution after the script terminates.
- 11 The script asks permission to overwrite three files in */usr/local/bin*: *dbhome*, *oraenv*, and *coraenv*.
- 12 The script restarts the Oracle CSS processes.

## Configuration

### Set the Environment

- Add the following code to *.bash\_profile*:

```
export
PATH=$PATH:/app/oracle/product/10.1.0/Db_2/bin
```
- Add the following code to *.bash\_profile*:

```
export
ORACLE_HOME=/app/oracle/product/10.1.0/Db_2
```

### Create a Database

This process is essentially the same as that described for the Windows platform. On Linux the Database Configuration Assistant binary is called *dbca* and is located in the *bin* directory that you just added to the *PATH* environment variable. For this exercise the SID name is *r10103*. The `ps -ef` output shows a number of processes whose names end with *\_r10103*.

You will notice that the `*:/app/oracle/product/10.1.0/Db_2:N` line in the */etc/oratab* has been changed to `r10103:/app/oracle/product/10.1.0/Db_2:N`. Modify this file to change the third field from *N* to *Y*. This will cause the *dbstart* utility to restart the database after a system reboot.

### Configure the Database

Run the *utlrp.sql* script.

- 1 Switch user to *oracle*.

- 2 Source the *oraenv* script: `. /usr/local/bin/oraenv`. The script prompts for values for ORACLE\_SID and ORACLE\_HOME. The output looks like:  
`ORACLE_SID = [oracle] ? r10103`  
`ORACLE_HOME = [/home/oracle] ? /app/oracle/product/10.1.0/Db_2`
- 3 `$ sqlplus "/ AS SYSDBA"`
- 4 `@?/rdbms/admin/utlrp.sql`

## Create a Listener

This process is essentially the same as that described for the Windows platform. On Linux the Net Configuration Assistant binary is called *netca* and is located in the *bin* directory that you just added to the PATH environment variable. The default name for the first listener is LISTENER. The `ps -ef` output displays a string that includes `bin/tnslsnr LISTENER -inherit`.

## Add a Service Name

Use *netca* to add the *r10103* service name. This process is very similar to the one described for the Windows platform.

## Automatic Startup of Oracle Database and Listener at System Boot Time

Put the following script called *init.oracle* in */etc/init.d*.

```
#!/bin/sh##### RHEL 3.0 and
Oracle 10.1.0# autostart oracle database and listen
er##### Comments to support
chkconfig on RedHat Linux# chkconfig: 2345 64 36# description:
starts and stops the database and the listenerORA_HOME=/app/or
acle/product/10.1.0/Db_2ORA_OWNER=oracleif [ ! -f /app/or
acle/product/10.1.0/bin/dbstart ]thenecho "Oracle Auto Startup:
CANNOT START"exitfi case "$1" in'start')su - oracle -c "/app/or
acle/product/10.1.0/DB_2/bin/dbstart &"su - oracle -c "/app/or
acle/product/10.1.0/DB_2/bin/lsnrctl start listener";;'stop')su -
oracle -c "/app/oracle/product/10.1.0/DB_2/bin/dbshut &"su - or
acle -c "/app/oracle/product/10.1.0/DB_2/bin/lsnrctl stop";;esac
```

---

**NOTE** Be sure that the oracle bin directory, for example, */app/oracle/product/10.1.0/Db\_2/bin*, has been added to the PATH variable for the oracle user account and that the ORACLE\_HOME variable has been set for the oracle user account. In this example the value of ORACLE\_HOME is */app/oracle/product/10.1.0/DB\_2*. Otherwise the listener start and listener stop calls fail with complaints about messages not found.

---

Run `chkconfig --add init.oracle`. `chkconfig` uses the information in the `chkconfig` comment line in *init.oracle* to set up calls to the script in the */etc/rc<n>.d* directories.

## Installing and Configuring an Oracle Client on Linux

### Introduction

What follows is a detailed description of the installation and configuration of a runtime version of an Oracle client on a linux x86 (RHEL3) machine. The runtime version is one of four installation types (instant, administrator, runtime, and custom).

# Installation

## Get the Installation Instructions

This installation involves many manual steps. You will need the installation instructions.

- 1 Load <http://www.oracle.com>. Click on Database under the Productslist header.
- 2 Select Database from the Documentationmenu.
- 3 Click Oracle Database 10g Release 1 (10.1) Documentationunder the Previous Releasesmenu.
- 4 Click PDF next to the B12091-02 part number. The document title is *Oracle Database Client Quick Installation Guide 10g Release 1 (10.1.0.3) for Linux x86*

## Get the Installer

Use the following information to get the installer used to install a Oracle 10g client on Linux.

**To get the installer:**

- 1 Go to <http://www.oracle.com/database/index.html>
- 2 Click FREE DOWNLOAD Download Oracle Database 10g Release 2
- 3 Click Oracle Database 10g Release 1 (10.1.0.3) Client for Linux x86
- 4 Click the checkbox labeled YES, I accept the License Terms and Export Restrictions and I acknowledge that I have reviewed and understood the agreement and agree to use the language I selected in entering into this agreement.
- 5 Click I Accept underneath the text box containing the licence agreement. This takes you to <http://www.oracle.com/technology/software/products/database/oracle10g/htdocs/linuxsoft.html>

- 6 Click `ship.client.lnx32.cpio.gz` which is located under the label Oracle Database 10g Client Release 1 (10.1.0.3) for Linux x86. This will download the file to your Linux machine into your home account.
- 7 `yourAccount> gunzip ship.client.lnx32.cpio.gz`
- 8 `yourAccount> cpio -idmv ship.client.lnx32.cpio`
- 9 `yourAccount> su - root`
- 10 `root> mkdir -p /app/oracle`
- 11 `root> chown -R oracle:oinstall /app/oracle`
- 12 `root> chmod -R 755 /app/oracle`
- 13 `root> chown -R oracle:oinstall Disk1`
- 14 `root> chmod -R 755 Disk1`
- 15 `root> mv Disk1 /home/oracle/`
- 16 Exit from root
- 17 `root> su - oracle`
- 18 `oracle> ORACLE_BASE=/app/oracle`
- 19 `oracle> export ORACLE_BASE`
- 20 `oracle> TEMP=/tmp`
- 21 `oracle> TMPDIR=/tmp`
- 22 `oracle> export TEMP TMP`
- 23 `oracle> unset ORACLE_HOME`
- 24 `oracle> unset TNS_ADMIN`

## Preliminary Steps

The steps for checking the software and hardware requirements, for creating the required UNIX group and user, for setting up the Oracle base directory, and configuring the Oracle user's environment are the same as for installing an Oracle database.

## Run the Installer

1 `oracle> Disk1/runInstaller`

2 Click Next to go to the Specify File Locations screen.

---

**NOTE** The source is `/home/oracle/Disk1/stage/products.xml`, and the destination is `/app/oracle/OraHome_1`.

---

3 Click Next to go to the Select Installation Type screen. For this exercise the Runtime installation type was selected.

4 Click Next to go to the Summary screen. Review the list of products that will be installed.

5 Click Install to go to the Install screen. The installer shows progress on this screen.

6 The installer ran the Oracle Net Configuration Assistant. Click Finish.

---

**NOTE** This step creates a `sqlnet.ora` file where the value (TNSNAMES, EZConnect) is assigned to the NAMES.DIRECTORY\_PATH parameter.

---

7 The installer halted to request that a script be run as root. The dialog displayed the following message. A configuration script needs to be run as root before installation can proceed. Leaving this window open, open another window and run `/app/oracle/oraHome_1/root.sh` as root, then return to this window and click OK to continue. This script writes three files, `dbhome`, `oraenv`, and `coraenv` to `/usr/local/bin`. It also writes an entry in `/etc/oratab`. This entry specifies that no databases should be brought up at system reboot time.

8 The installer was authorized to continue, and it completed.

## Configuration

### Set Environment Variables

Set and export the ORACLE\_HOME variable. The value in this exercise is `/app/oracle/OraHome_1`. Add the `bin` directory to the PATH variable.

## Add a Local Name Service

Run the Oracle Net Configuration Assistant. The name of the executable is *netca*, and it is in the *bin* directory. The assistant prompts you to define a service name, a network protocol, a host name, and a port. With this information, it creates an entry for the service name in the *tnsnames.ora* file

- 1 Click the radio button labeled Local Net Service Name Configuration. Click Next.
- 2 The add operation is selected by default. Click Next.
- 3 In the text box labeled Service Name type the Global Database Name of the target Oracle instance. For this exercise the service name is *r10102.ads.autodesk.com*. Click Next.
- 4 The TCP protocol is selected by default. Click Next.
- 5 In the text box labeled Hostname type the fully qualified name of the host where the Oracle instance is located. The port value is 1521 by default. Click Next.
- 6 The option to test the connection is selected by default. Click Next.
- 7 By default the text box labeled Net Service Name contains the root of the service name, that is, the characters preceding the dot (.) signaling the start of the domain name part. For this exercise this value is *r10102*. Click Next.
- 8 The option not to configure another name is selected by default. Click Next.

---

**NOTE** These steps create a *tnsnames.ora* file in *\$ORACLE\_HOME/network/admin*. This file contains an entry of the form *<Net Service Name> = (...)*. In the entry are values defined for protocol, host, port, and service name taken from the input you gave.

---

## Testing the Connection

The connection is made using *sqlplus* between an Oracle instance on an XP machine and an Oracle client on a RHEL3 Linux machine. The XP machine is protected by a corporate firewall, which allows inbound connections on port 1521. The behavior of Oracle was modified using *regedit* to cause all traffic including connection requests to go through port 1521. In *regedit* create



a new string value whose name is `USE_SHARED_SOCKET` and whose value is `TRUE`. Create this value in the folder *My Computer\HKEY\_LOCAL\_MACHINE\SOFTWARE\ORACLE\KEY\_OraDb10g\_home1*. The folder name *KEY\_OraDb10g\_home1* may be different on your machine.

The sqlplus command is as follows.

```
SQL> connect sys/test@R10102 as sysdba
```

```
Connected.
```

```
SQL>
```

'R10102' is the net service name in the `tnsnames.ora` file. 'test' is the password for the 'sys' account in the Oracle instance on the XP machine.

## Installing and Configuring SQL Server 2000

### Installation

This describes the installation of SQL Server client tools on a Windows XP system. Do the following.

- 1 Insert the Microsoft SQL Server 2000 Enterprise Edition CD into the drive.
- 2 Click on SQL Server 2000 Components.
- 3 Click on Install Database Server.

---

**NOTE** A warning dialog will appear. Microsoft SQL Server 2000 Enterprise server component is not supported on this operating system. Only client components will be available for installation.

---

- 4 Use the default name for the computer (local computer).
- 5 Use the default action (Create a new instance of SQL Server or install Client Tools).
- 6 Enter your name.
- 7 Enter 25-digit CD-key.
- 8 Accept the default list of components to install.

## Configuration

Use the Enterprise Manager to add a SQL Server to a SQL Server group. Do the following.

- 1 Click Enterprise Manager in the Microsoft SQL Server menu. The menu path is Start ► Programs ► Microsoft SQL Server.
- 2 Click the + to expand the node labeled Microsoft SQL Servers in the Console Root pane.
- 3 Click New SQL Server Registration from the pop-up menu accessed by right-clicking on the SQL Server Group node in the Console Root pane. As a result the Register SQL Server Wizard window is displayed. Click Next.
- 4 Select a server from the list of available servers and click Add. Click Next.
- 5 Use the default Windows Authentication. Click Next.
- 6 Accept the default. This adds the server that you selected to the default server group called SQL Server Group. Click Next. Click Finish. A dialog is displayed saying that the registration has been completed.

## Connection

The name of the SQL Server that you added is the value that you will use for the Service connection property. You can connect to the SQL Server that you added by just setting the Service connection property because you specified Windows Authentication during the configuration procedure. The connection will not succeed unless your windows account name has been added as a new login to the remote server.

# Installing and Configuring SQL Server 2005 Developer Edition

## Installation

This topic describes installation of the Developer Edition from a CD. You must install .NET Framework 2.0 first. Then do the following.

---

**NOTE** During initialization the installer complained that it could not find an application to open the *splash.hta* file. A google on 'splash.hta' yielded the suggestion to try *C:\WINDOWS\system32\mshta.exe*. This worked.

---

- 1 Accept the EULA.
- 2 Select the components to install. For this exercise database services, analysis services, and workstation components were installed. Reporting Services, Notification Services, and Integration Services were not.
- 3 Select the instance name. The default (MSSQLSERVER) was chosen.
- 4 Select a service account. A domain user account was supplied.
- 5 Select the services to be started upon completion of the setup. SQL Server and Analysis Services were chosen.
- 6 Select the authentication mode. Windows authentication mode was selected.
- 7 Select the collation setting. The default SQL Collation was left selected. This is described as dictionary order, case-insensitive, for use with 1252 Character Set.
- 8 Install the components. The components installed are:
  - MSXML6
  - SQL Server Setup Support Files
  - SQL Server Native Client
  - SQL VSS Writer (service for backup and restore)
  - OWC11 (a component for publishing database content to the web)
  - Backward compatibility files

- Database Services
- Analysis Services
- SQLXML4
- SQL Server Books Online
- Workstation Components

---

**NOTE** The first installation failed to install all of the expected tools. The second installation succeeded after uninstalling the SQL Server Mobile Edition that had been installed by the Visual Studio 2005 installer.

---

## Connection

Use the SQL Server Management Studio to connect to a SQL Server. Before you can connect, your database administrator must add your Windows account to the server as a login account. Do the following.

- 1 Click SQL Server Management Studio in the Microsoft SQL Server 2005 menu. The menu path is Start ► Programs ► Microsoft SQL Server 2005.
- 2 The Management Studio window is displayed with a Connect to Server dialog in the foreground.
- 3 There are three spinboxes labeled Server type, Server name, and Authentication. The server type was set to Database Engine. The authentication was set to Windows Authentication. The server name had to be set to that of a known SQL Server 2005 server.
- 4 Click Connect. The result is that the components of the named server are displayed in the Object Explorer.

# The Open Source Providers

# 4

This chapter describes how to connect to and configure the open source providers (ArcSDE, MySQL, ODBC, SDF, SHP, WFS, and WMS).

## System Requirements

### Software

The following table maps the various providers to the operating system and RDBMS versions against which they have been tested (T). The RDBMS-based providers (ArcSDE and MySQL) require that the corresponding RDBMS client is installed. The ODBC provider requires that an ODBC driver be installed on Linux. Windows XP contains the required ODBC drivers.

FDO and its Providers run on the Windows Vista operating system. However, Autodesk cannot guarantee that the underlying file or database that an FDO Provider is connecting to is Vista compatible (except for the Autodesk Spatial Data File format, SDF). Please consult the file or database vendor to verify Vista compatibility.

**WXP=Windows XP Pro; W2K=Windows 2000 Pro; W2K3=Windows 2003 Server**

Provider	WXP	W2K	W2K3	Linux	RDBMS	Notes
ArcSDE	T	T	T	T	ArcSDE 9.1 supported. SQL Server 2005 certified. SQL Server 2000, SP4 not certified. Oracle 10gR2	1

WXP=Windows XP Pro; W2K=Windows 2000 Pro; W2K3=Windows 2003 Server						
Provider	WXP	W2K	W2K3	Linux	RDBMS	Notes
					certified. Oracle 10gR1 and Oracle 9i not certified.	
MySQL	T	T	T	T	Latest MySQL version with MyIsam certified; earlier versions with MyIsam should work.	2
ODBC	T	T	T	T	Supports read/write of data and geometry (where there is no limitation by the source itself) on connections to Access, Excel, SQL Server, Oracle and MySQL. Cannot delete in Excel.	3
SDF	T	T	T	T	n/a	4
SHP	T	T	T	T	n/a	
WFS	T	T	T	T	n/a	5
WMS	T	T	T	T	n/a	6

#### NOTE

- 1 The ArcSDE 9.1 client and ArcSDE 9.1 server for Oracle and ArcSDE 9.1 server for SQL Server are required.
- 2 Testing the MyIsam database engine with spatial extension only.
- 3 The MDAC 2.8 ODBC driver manager is tested on Windows for access to all of the named data sources. The intent is to test the new Oracle ODBC driver for read-only access on Linux in the near future.
- 4 File formats 3.0 and 3.2 are supported. An SDF file can have only one schema.
- 5 WFS 1.0.0 and GML 2.1.2
- 6 WMS 1.1.0, WMS 1.1.1, WMS 1.3, and GML 2.1.2

## Disk Space and Memory

This table shows the disk space/memory requirements the RDBMS clients on an XP machine.

### Windows

This table shows the disk space/memory requirements for the RDBMS clients.

**D = Disk (MB); M = Memory (MB); N = Notes**

Component	D	M	N
ArcSDE 9.1 client (3 DLLs)	26.3	4	1
MySQL 5.0 Community Edition (mysqld-nt.exe)	94	13	2

#### NOTE

- 1 The ArcSDE client disk size is reflected in the total, but the memory size would be included in that of the running FDO application.
- 2 MySQL includes both server and client components. There is no separate client.

### Linux

This table shows the disk space/memory requirements for the RDBMS clients.

**D = Disk (MB); M = Memory (MB); N = Notes**

Component	D	M	N
ArcSDE 9.1 client	26.4	4	1
MySQL 5.0 Community Edition (mysqld)	49.6	1.25	2

---

**NOTE**

- 1 The ArcSDE client disk size is reflected in the total, but the memory size would be included in that of the running FDO application.
  - 2 MySQL includes the following RPMs: MySQL-client-standard, MySQL-devel-standard, MySQL-server-standard, and MySQL-shared-standard.
- 

## Feature Matrix

The main features for Commands, Schema Capabilities, and Tools and Setup of the FDO API are listed in the table . “Y” means the command or attribute is supported in that particular provider.

---

**NOTE** The features are described in the [Feature Description](#) (page 7) section.

---

Feature	Arc-SDE	MySQL	ODBC	SDF	SHP	WFS	WMS	GDAL
<b>Commands</b>								
Connection	Y	Y	Y	Y	Y	Y	Y	Y
List Datastores	Y	Y						
Create Datastore		Y	Y	Y				
Destroy Datastore		Y	Y	Y				
Describe Schema	Y	Y	Y	Y	Y	Y	Y	Y
Apply Schema		Y		Y	Y			
Destroy Schema		Y			Y			
Get Spatial Contexts	Y	Y	Y	Y	Y	Y	Y	Y



Feature	Arc-SDE	My SQL	ODBC	SDF	SHP	WFS	WMS	GDAL
Create Spatial Context	Y	Y		Y	Y			
Destroy Spatial Context	Y	Y						
Select	Y	Y	Y	Y	Y	Y	Y	Y
Select Aggregates	Y	Y	Y	Y	Y	Y	Y	Y
Insert	Y	Y	Y	Y	Y			
Update	Y	Y	Y	Y	Y			
Delete	Y	Y	Y	Y	Y			
SQL Command	Y	Y	Y		Y			
Transactions	Y							
Locking	Y							
Long Transactions	Y							
<b>Schema Capabilities</b>								
Schema Overrides		Y	Y	Y		Y	Y	
Foreign Schemas		Y	Y					
Associations	Y	Y		Y				
Data Property Constraints		Y		Y	Y			
Inheritance		Y	Y	Y	Y	Y	Y	Y

Feature	Arc-SDE	My SQL	ODBC	SDF	SHP	WFS	WMS	GDAL
Object Properties		Y				Y		
<b>Setup and Tools</b>								
Feature Schema Serialization	Y	Y	Y	Y	Y	Y	Y	Y
Feature Data Serialization	Y	Y	Y	Y	Y	Y	Y	Y
XML Configuration			Y				Y	Y

## Connection API

The following table shows the connection properties for each provider. “T” means true, and “F” means false. For important details, refer to the associated Notes.

**DV=Default Value; R=Required; P=Protected; FN=FileName; FP=FilePath; D=Datastore; E=Enumerable**

Provider/Properties	DV	R	P	FN	FP	D	E	Notes
OSGeo.ArcSDE.3.0								
Server		T	F	F	F	F	F	1
Instance		T	F	F	F	F	F	2
Username		T	F	F	F	F	F	3,4
Password		T	T	F	F	F	F	3,4
Datastore		F	F	F	F	T	T	5,6

**DV=Default Value; R=Required; P=Protected; FN=FileName; FP=FilePath; D=Datastore; E=Enumerable**

Provider/Properties	DV	R	P	FN	FP	D	E	Notes
OSGeo.MySQL.3.0								7
Username		T	F	F	F	F	F	8
Password		T	T	F	F	F	F	8
Service		T	F	F	F	F	F	9
DataStore		F	F	F	F	T	T	10
OSGeo.ODBC.3.0								11
DataSourceName		F	F	F	F	F	T	12
UserId		F	F	F	F	F	F	13
Password		F	T	F	F	F	F	13
ConnectionString		F	F	F	F	F	F	14
GenerateDefaultGeometryProperty	True	F	F	F	F	F	F	
OSGeo.SDF.3.0								15
File		T	F	T	F	F	F	16
ReadOnly	False	F	F	F	F	F	T	17
OSGeo.SHP.3.0								
DefaultFileLocation		F	F	T	F	F	F	18

DV=Default Value; R=Required; P=Protected; FN=FileName; FP=FilePath; D=Datastore; E=Enumerable

Provider/Properties	DV	R	P	FN	FP	D	E	Notes
TemporaryFileLocation		F	F	T	F	F	F	19
OSGeo.WFS.3.0								
FeatureServer		T	F	F	F	F	F	20
Username		F	F	F	F	F	F	
Password		F	F	F	F	F	F	
OSGeo.WMS.3.0								21
FeatureServer		T	F	F	F	F	F	
Username		F	F	F	F	F	F	
Password		F	F	F	F	F	F	
DefaultImageHeight		F	F	F	F	F	F	

---

**NOTE**

- 1 The value of the Server property is the name or IP address of the host where the ArcSDE 9.1 Server instance runs.
- 2 The value of the Instance property defines the port and protocol to use when communicating with the ArcSDE server. This instance name must be mapped to the TCP/IP listen port used by the Server to accept connection requests. This mapping is stored in the `C:\WINDOWS\system32\drivers\etc\services` file and has the form `"instance_name portnum/protocol # comment"`. For example, `"esri_sde_oracle 5151/tcp #ArcSDE Server listening port"`. There can be multiple instances because the server can connect to multiple back-end database technologies, for example, both Oracle and SQL Server.
- 3 You must create the username and password from the `sqlplus` command-line when FDO Provider for ArcSDE is using Oracle as the back-end database technology. This is described in topic [Create a Username and Password for the ArcSDE Provider Connected to Oracle](#) (page 81).
- 4 You must create the username and password from the Enterprise Manager when FDO Provider for ArcSDE is using SQL Server as the back-end database technology. This is described in topic [Create a Username and Password for the ArcSDE Provider Connected to SQL Server](#) (page 81).
- 5 When FDO Provider for ArcSDE is using Oracle as the back-end database, the value of DataStore is always 'Default Datastore'.
- 6 When FDO Provider for ArcSDE is using SQL Server as the back-end database, the value of DataStore is one of the values in the list of databases supported by that instance of the SQL Server.
- 7 During the Windows installation of MySQL, be sure to specify a password for the root account.
- 8 See [User Management API](#) (page 18) for information about adding a username and password.
- 9 The value of the Service property is `localhost` for a local connection and a valid domain name for a remote connection. A firewall at the remote end can prevent a connection from succeeding. However, note that the exception message will not tell you that this is the cause.
- 10 The value of the DataStore property is one of the values enumerated in the connection property dictionary after opening a pending connection

to the provider. This is described in topic [Open a Pending Connection](#) (page 16).

- 11 You cannot use the ODBC provider either to create or destroy a feature schema. The provider will interpret numeric fields named X and Y to contain coordinate data; otherwise you must supply it with an xml file that tells it which column is X and which column is Y. With one qualification, you can use the provider to insert, update, and delete features in the data store; you cannot delete features in an Excel-based data store. See the topic Overview of the Connection Properties for an overview of the use of ODBC connection properties.
- 12 Use a Windows OS administrative tool to define a User or System DSN (data source name). FDO does not support the use of File DSN. This action associates a symbolic name with a resource that specifies how to access the data store, for example, a file path or RDBMS connection properties. Adding a User DSN for a Microsoft Access database file is described in the topic [Create an ODBC DSN for an Excel File](#) (page 84). Adding a User DSN for a Microsoft Excel file is described in the topic [Create an ODBC DSN for an Excel File](#) (page 84). Adding a User DSN for an Oracle database is described in the topic [Create an ODBC DSN for an Oracle Database](#) (page 85).
- 13 Optional properties, depending on the data source.
- 14 The connection string has the format  
`DSN=myDSN;Uid=myUsername;Pwd=myPassword; OR DSN=myDSN;Uid=;Pwd=;`  
if there is no username or password.
- 15 The SDF provider supports the `.sdf` file type.
- 16 This is a file name. The named file is in the local file system.
- 17 If True, only read operations on the file are permitted.
- 18 The DefaultFileLocation can be either a directory in the file system or the name of an existing file.
- 19 The TemporaryFileLocation is a folder where the SHP provider creates files that are normally co-located with the `.shp` file named in the DefaultFileLocation property, but are, for some reason, missing. These files have file types `.shx`, `.sbn`, `.idx`, `.dbf`, and `.sbx`.
- 20 The value assigned to the FeatureServer property is a URL that identifies the location of a WFS server, for example,  
<http://regis.intergraph.com/wfs/dcmetro/request.asp>.

- 21 The value assigned to the FeatureServer property is a URL that identifies the location of a WMS server, for example, <http://wms.jpl.nasa.gov/wms.cgi>. A WMS server may support more than one version of the OGC WMS specification. The FDO WMS provider allows users to connect to a WMS server that support the following WMS versions: 1.1.0, 1.1.1, and 1.3.0. By default, the FDO WMS provider will use version 1.3.0 when connecting to a WMS server. If a user wishes to change the default behavior and override the version that the provider uses to connect to the server, the value assigned to the FeatureServer connection property can be modified to include a version specification. This may be necessary if a server does not support a client connecting to the server using a version number of 1.3.0. The following are valid examples on how to explicitly specify the version number: <http://terraservice.net/ogccapabilities.ashx?version=1.1.1>, and [http://www.bsc-eoc.org/cgi-bin/bsc\\_ows.asp?version=1.1.0](http://www.bsc-eoc.org/cgi-bin/bsc_ows.asp?version=1.1.0). If version information is specified in such a manner, the WMS provider will honor the request and use the specified version.
- 

## Create a Username and Password for the ArcSDE Provider Connected to Oracle

Do the following:

- 1 `sql> connect sys/password as sysdba`
- 2 `sql> create user username identified by password;`
- 3 `sql> grant connect, resource, select any table, insert any table, update any table, delete any table to username;`

## Create a Username and Password for the ArcSDE Provider Connected to SQL Server

The following procedure adds a new login user named *name* with password *password* who must login to the SQL Server using SQL Server authentication. The user named *name* is given the global privilege, Database Creators, and is

added to a database with privileges, db\_ddladmin, db\_datareader, and db\_datawriter.

- 1 On the Windows Start menu, click Programs, click Microsoft SQL Server, click Enterprise Manager.
- 2 In the Enterprise Manager application window, click Console Root, click Microsoft SQL Servers, click SQL Server Group, click *GroupName*, click Security, click Logins, click New Login. The SQL Server Login Properties - New Login dialog box is displayed.
- 3 In the SQL Server Login Properties - New Login dialog box, click the tab labeled General, type *name* in the text entry box labeled Name, click the radio button labeled SQL Server Authentication, type a *password* in the text entry box labeled Password.
- 4 Click the tab labeled Server Roles, click the check-box labeled Database Creators
- 5 Click the tab labeled Database Access, click the check-box in the column labeled Permit next to the name of the database to which you are adding this user, click the check-boxes in the column labeled Permit in Database Role next to the labels db\_ddl\_admin, db\_datareader, and db\_datawriter.
- 6 Click OK. The Confirm Password dialog box is displayed.
- 7 In the Confirm Password dialog box, type *password* in the text entry box labeled Password, click OK.
- 8 In the SQL Server Login Properties - New Login dialog box, click OK.

## Connecting to the ODBC Provider

### Overview of the Connection Properties

The simplest case is connecting to a file-based data store. You create a DSN for the file as described in the following topics for MS Access and Excel, set the DataSourceName connection property to that DSN and then connect. Alternatively, you can just set the DSN parameter in the ConnectionString connection property, for example, `DSN=ExcelDSN;Uid=;Pwd=.`

The more complicated cases concern connection to an RDBMS-based provider.



When you use the Windows administrative tool to configure the DSN for an Oracle connection, you must define a Data Source Name and a TNS Service Name. You can optionally define a User ID. In the event that you do not define a User ID when configuring the DSN, you can connect by either setting the DataSourceName, UserId, and Password connection properties or by setting the DSN, Uid, and Pwd parameters in the ConnectionString connection property. In the event that you did define a User ID as part of the DSN, can connect by either setting the DataSourceName and Password connection properties or by setting the DSN and Pwd parameters in the ConnectionString connection property. The User ID defined in the DSN or the user name value defined in the connection property is used as the default schema for the session; that is, you will see tables from that schema.

When you use the Windows administrative tool to configure the DSN for a MySQL connection, you must define the Data Source Name, the Server, and the Database. Before connecting you must define the DataSourceName, the UserId, and Password connection properties. Alternatively, you must define the DSN, Uid, and Pwd parameters in the ConnectionString connection property.

When you use the Windows administrative tool to configure the DSN for a SQL Server connection, you must define the Name and Server. The default authentication is to use the Windows account name and password. You can change the default database from master to one of your own. Before connecting you must define the DataSourceName connection property.

## Create an ODBC DSN for an MS Access File

Do the following:

- 1 On the Windows Start menu, click Settings, click Control Panel, click Administrative Tools, click Data Sources (ODBC). The ODBC Data Source Administrator dialog box is displayed.
- 2 In the Data Source Administrator dialog box, click the User DSN tab if you want the data source to be visible only to you on this machine, or click the System DSN tab if you want the data source to be visible to others on this machine and on the network. Click Add. The Create New Data Source dialog box is displayed.

---

**NOTE** You may not use the File DSN tab, because Fdo does not support its use.

---

- 3 In the Create New Data Source dialog box, click Microsoft Access Driver (\*.mdb), click Finish. The ODBC Microsoft Access Setup dialog box is displayed.
- 4 In the ODBC Microsoft Access Setup dialog box, in the text entry box labeled Data Source Name , type the *name* for the data source that you want to use as the value of the DSN parameter in the connection string, click Select. The Select Database dialog box is displayed.
- 5 In the Select Database dialog box, browse to the folder where the MS Access file is, select the file, and click OK.
- 6 In the ODBC Microsoft Access Setup dialog box, click OK.
- 7 In the Data Source Administrator dialog box, click OK.

## Create an ODBC DSN for an Excel File

Do the following:

- 1 On the Windows Start menu, click Settings, click Control Panel, click Administrative Tools, click Data Sources (ODBC). The ODBC Data Source Administrator dialog box is displayed.
- 2 In the Data Source Administrator dialog box, click the User DSN tab if you want the data source to be visible only to you on this machine, or click the System DSN tab if you want the data source to be visible to others on this machine and on the network. Click Add. The Create New Data Source dialog box is displayed.

---

**NOTE** You may not use the File DSN tab, because Fdo does not support its use.

---

- 3 In the Create New Data Source dialog box, click Microsoft Access Driver (\*.mdb), click Finish. The ODBC Microsoft Access Setup dialog box is displayed.
- 4 In the ODBC Microsoft Access Setup dialog box, in the text entry box labeled Data Source Name , type the *name* for the data source that you want to use as the value of the DSN parameter in the connection string, click Select. The Select Database dialog box is displayed.
- 5 In the Select Database dialog box, browse to the folder where the MS Access file is, select the file, and click OK.

- 6 In the ODBC Microsoft Access Setup dialog box, click OK.
- 7 In the Data Source Administrator dialog box, click OK.

## Create an ODBC DSN for an Oracle Database

Do the following:

- 1 On the Windows Start menu, click Settings, click Control Panel, click Administrative Tools, click Data Sources (ODBC). The ODBC Data Source Administrator dialog box is displayed.
- 2 In the ODBC Data Source Administrator dialog box, click the User DSN tab if you want the data source to be visible only to you on this machine, or click the System DSN tab if you want the data source to be visible to others on this machine and on the network. Click Add. The Create New Data Source dialog box is displayed.

---

**NOTE** You may not use the File DSN tab, because Fdo does not support its use.

---

- 3 In the Create New Data Source dialog box, click Oracle in <ORACLE\_HOME\_NAME>, click Finish. The Oracle ODBC Driver Configuration dialog box is displayed.

---

**NOTE** <ORACLE\_HOME\_NAME> represents the value of the Windows registry ORACLE\_HOME\_NAME parameter. See HKEY\_LOCAL\_MACHINE/SOFTWARE/ORACLE/KEY\_<ORACLE\_HOME\_NAME>

---

- 4 In this dialog box, in the text entry box labeled Data Source Name , type the *name* that you want to use as the value of the DataSourceName connection property.
- 5 In this dialog in the text entry box labeled TNS Service Name, type the <SERVICE\_NAME> found in the tnsnames.ora file.

---

**NOTE** For example, here's a typical path to the tnsnames.ora file:  
C:\oracle\product\10.1.0\Db\_1\NETWORK\ADMIN. In this file there are entries similar to the following: ORCL = (..(CONNECT\_DATE = (...(SERVICE\_NAME = orcl)))

---

- 6 Providing a value for UserId parameter is optional. In this dialog in the text entry box labeled User ID type in **all uppercase** the name of an Oracle user. Click Test Connection. The Oracle ODBC Driver Connect dialog appears. The Service Name and User Name text entry boxes are populated. Type the password in the text box labeled Password. Click Test Connection. The Testing Connection dialog appears, saying that the connection is successful. Click OK to close this dialog. Click OK to close the Configuration dialog.

---

**NOTE** If you type the User ID in all uppercase letters, you will be able to use all lowercase or all uppercase letters for the UserID connection property value; otherwise you will have to use all uppercase for that value.

---

- 7 In the ODBC Data Source Administrator dialog box, you see the DSN that you just added listed. Click OK.

## Create an ODBC DSN for a MySQL Database

Do the following:

- 1 On the Windows Start menu, click Settings, click Control Panel, click Administrative Tools, click Data Sources (ODBC). The ODBC Data Source Administrator dialog box is displayed.
- 2 In the ODBC Data Source Administrator dialog box, click the User DSN tab if you want the data source to be visible only to you on this machine, or click the System DSN tab if you want the data source to be visible to others on this machine and on the network. Click Add. The Create New Data Source dialog box is displayed.

---

**NOTE** You may not use the File DSN tab, because Fdo does not support its use.

---

- 3 In the Create New Data Source dialog box, click MySQL ODBC 3.51 Driver, click Finish. The Connector/ODBC 3.51.12 dialog box is displayed.
- 4 In this dialog box, in the text entry box labeled Data Source Name , type the *name* that you want to use as the value of the DataSourceName connection property.

- 5 In this dialog in the text entry box labeled Server, type the **hostname/IP address** of the machine where the MySQL database server is installed. The default is localhost.
- 6 You can ignore text entry boxes labeled User and Password. You provide that information using the Fdo UserId and Password connection properties.
- 7 Pick a database using the drop-down combination box labeled Database.
- 8 Click OK to close the Configuration dialog.
- 9 In the ODBC Data Source Administrator dialog box, you see the DSN that you just added listed. Click OK.

## Create an ODBC DSN for a SQL Server Database

Do the following:

- 1 On the Windows Start menu, click Settings, click Control Panel, click Administrative Tools, click Data Sources (ODBC). The ODBC Data Source Administrator dialog box is displayed.
- 2 In the ODBC Data Source Administrator dialog box, click the User DSN tab if you want the data source to be visible only to you on this machine, or click the System DSN tab if you want the data source to be visible to others on this machine and on the network. Click Add. The Create New Data Source dialog box is displayed.

---

**NOTE** You may not use the File DSN tab, because Fdo does not support its use.

---

- 3 In the Create New Data Source dialog box, click SQL Server, click Finish. The Create a New Data Source to SQL Server dialog box is displayed.
- 4 In this dialog box, in the text entry box labeled Name , type the *name* that you want to use as the value of the DataSourceName connection property.
- 5 In this dialog in the text entry box labeled Server, type the **name** of the SQL Server that you want to connect to.
- 6 Click Next to go to the next screen where you see the network login Id is used to verify the user's identity.

- 7 Click Next to go to the next screen where you can change the default database.
- 8 Click Next to go to the next screen where you can change other parameters.
- 9 Click Finish. The ODBC Microsoft SQL Server Setup dialog appears showing the configuration parameter settings.
- 10 In this dialog click Test Data Source.... The SQL Server ODBC Data Source Test dialog appears giving you test results. Click OK to close this dialog.
- 11 Click OK to close the ODBC Microsoft SQL Server Setup dialog .
- 12 In the ODBC Data Source Administrator dialog box, you see the DSN that you just added listed. Click OK.

## Command Capabilities

The list of command class names is shown in the first table. For descriptions of what the commands do, see the API reference documentation. The command attributes are shown in the second table. For descriptions of what the attributes mean, see entry for the `FdoICommandCapabilities` class in the API reference documentation. “Y” means the command or attribute is supported in that particular provider.

Command Class Name	Arc-SDE	MySQL	ODBC	SDF	SHP	WFS	WMS
<code>FdoISelect</code>	Y		Y	Y	Y	Y	Y
<code>FdoISelectAggregates</code>	Y	Y	Y	Y	Y		Y
<code>FdoIInsert</code>	Y	Y	Y	Y	Y		
<code>FdoIDelete</code>	Y	Y	Y	Y	Y		
<code>FdoIUpdate</code>	Y	Y	Y	Y	Y		
<code>FdoIDescribeSchema</code>	Y	Y	Y	Y	Y	Y	Y

Command Class Name	Arc-SDE	MySQL	ODBC	SDF	SHP	WFS	WMS
FdoIDescribeSchemaMapping			Y		Y		Y
FdoIApplySchema		Y		Y	Y		
FdoIDestroySchema		Y			Y		
FdoIActivateSpatialContext	Y	Y					Y
FdoICreateSpatialContext	Y	Y		Y			
FdoIDestroySpatialContext	Y	Y					
FdoIGetSpatialContexts	Y	Y	Y	Y	Y	Y	Y
FdoICreateDataStore		Y		Y			
FdoIDestroyDataStore		Y		Y			
FdoIListDataStores		Y					
FdoICreateSDFFile				Y			
FdoISQLCommand	Y	Y	Y				
FdoIAcquireLock	Y						
FdoIGetLockInfo	Y						
FdoIGetLockedObjects	Y						
FdoIGetLockOwners	Y						
FdoIReleaseLock	Y						

Command Class Name	Arc-SDE	MySQL	ODBC	SDF	SHP	WFS	WMS
FdoIActivateLongTransaction	Y						
FdoICommitLongTransaction	Y						
FdoICreateLongTransaction	Y						
FdoIDeactivateLongTransaction	Y						
FdoIFreezeLongTransaction							
FdoIGetLongTransactions	Y						
FdoIRollbackLongTransaction	Y						
FdoICreateSpatialIndex		Y					
FdoIDestroySpatialIndex		Y					
FdoIGetSpatialIndexes		Y					

**NOTE** The capability to create and destroy a data store is associated with the capability to create and destroy a feature schema. To create a data store you specify values for data store properties. For more information about these properties are discussed in [Data Store Management API](#) (page 19).

Attribute	Arc-SDE	MySQL	ODBC	SDF	SHP	WFS	WMS
Parameters	Y						
Timeout							
SelectExpressions		Y	Y	Y	Y		



Attribute	Arc-SDE	MySQL	ODBC	SDF	SHP	WFS	WMS
SelectFunctions	Y	Y	Y	Y	Y		
SelectDistinct	Y	Y	Y	Y	Y		
SelectOrdering	Y	Y	Y		Y		
SelectGrouping		Y	Y				

## User Management

In the open source release there is no API for this functional domain. Before you can connect to your own database account using MySQL or ArcSDE provider, someone with the requisite privileges must create that account for you. The following topics show the commands used to create accounts and perform other user-related operations.

### MySQL

The following table maps a set of user-related operations expressed as pseudo-API calls to the SQL commands that perform them.

Operation	MySQL	Notes
AddUser(name, password)	create user <i>name</i> identified by ' <i>password</i> '	1
AssignRoleOrPrivilege(user, roleOrPriv)	grant <i>roleOrPriv</i> on *.* to <i>user</i>	2
DropUser(name)	drop user <i>name</i>	
GetDatastores()	show databases	3
GetDbUsers()	select user as name from mysql.user	4

Operation	MySQL	Notes
GetDbUsers(inDatastore)	select distinct grantee as name from information_schema.schema_privileges where table_schema = ' <i>inDatastore</i> '	5
GetRolesOrPrivileges()	select distinct privilege_type from information_schema.user_privileges where grantee like "'root'%" order by privilege_type	6
GrantAccessToDatastore(user, datastore)	grant all on <i>datastore</i> .* to <i>user</i>	7
GrantAccessToDatastore(user, datastore, priv)	grant <i>priv</i> on <i>datastore</i> .* to <i>user</i>	8
RevokeAccessToDatastore( user, datastore, priv)	revoke <i>priv</i> on <i>datastore</i> .* from <i>user</i>	9
RevokeRoleOrPrivilege(user, roleOrPriv)	revoke <i>roleOrPriv</i> on *.* from <i>user</i>	10
ShowAccessToDatastore(user, datastore)	select privilege_type from information_schema.schema_privileges where table_schema = ' <i>datastore</i> ' and grantee = "' <i>user</i> @'%" order by privilege_type	
ShowPrivileges(user)	select * from mysql.user where user = ' <i>user</i> '	11

---

**NOTE**

- 1 You must have the global CREATE USER privilege or the INSERT privilege for the mysql database. A new MySQL user is created with no privileges. Assume that you have created a new user called newuser on the MySQL instance on your machine. Execute `mysql> show grants for newuser;`. The result string is "GRANT USAGE ON \*.\* TO 'newuser'@'%' IDENTIFIED BY PASSWORD '\*94BDCEBE19083CE2A1F959FD02F964C7AF4CFC29'". This means that newuser has no privileges and can only log in locally. To give the user default privileges in a datastore execute the SQL command that maps to the pseudo-API call `GrantAccessToDatastore(user, datastore)`.
- 2 To confirm the results of this command, execute the SQL command that maps to the pseudo-API call `ShowPrivileges(user)`.
- 3 You can get the same information from the `FdoListDatastores` command.
- 4 If you are connected to MySQL and you created the user with `AddUser`, USAGE is shown as the privilege.
- 5 All users in a particular data store are listed along with the privileges that they have in the data store.
- 6 The list of supported privileges for MySQL consists of ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE USER, CREATE VIEW, DELETE, DROP, EXECUTE, FILE, INDEX, INSERT, LOCK TABLES, PROCESS, REFERENCES, RELOAD, REPLICATION CLIENT, REPLICATION SLAVE, SELECT, SHOW DATABASES, SHOW VIEW, SHUTDOWN, SUPER, and UPDATE.
- 7 This command grants a set of default privileges to the user when connected to that data store. The privileges are alter, alter routine, create, create routine, create temporary tables, create view, delete, drop, execute, index, insert, lock tables, references, select, show view, and update. To verify the result execute the SQL command that maps to the pseudo-API call `ShowAccessToDatastore(user, datastore)`.
- 8 This command grants the specified privilege to that user for the named data store. The MySQL provider does not allow you to grant any privilege other than those assigned by a call to `GrantAccessToDatastore(user, datastore)`. To verify the result execute the SQL command that maps to the pseudo-API call `ShowAccessToDatastore(user, datastore)`.
- 9 This command revokes the specified privilege from that user for the named data store. If the role is set to '\*', the provider revokes all privileges in the data store from the user. To verify the result execute the SQL

command that maps to the pseudo-API call `ShowAccessToDatastore (user, datastore)`.

- 10 This command revokes the specified role or privilege. To verify the result execute the SQL command that maps to the pseudo-API call `ShowPrivileges (user)`.
- 11 This SQL command returns a list of boolean values indicating whether the user has the privilege.

---

## Data Store Management API

This table shows the data store management properties for each provider. “T” means true, and “F” means false.

DV=DefaultValue; R=Required; P=Protected; FN=FileName; D=Datastore; E=Enumerable							
Provider/Properties	DV	R	P	FN	D	E	Notes
OSGeo.MySQL.3.0							
DataStore		T	F	F	T	F	1
Description		F	F	F	F	F	

---

### NOTE

- 1 A data store is a container in the data source to which you can apply a feature schema.
- 

## Spatial Context API

The following table defines default values for the spatial context properties for the providers (MySQL and SDF) that support the creation of a data store

and example sets of spatial context properties extracted from ODBC, WFS, and WMS sources created by a non-FDO process.

Properties/Provider	MySQL	ODBC	SDF	WFS	WMS	Notes
Coordinate system	Cartesian	Cartesian	Cartesian	EPSG:4326	EPSG:4326	1
Extent type	Static	Static	Dynamic	Static	Static	2
Extent	Not NULL	Not NULL	NULL	Not NULL	Not NULL	3
Unit dimension	XY	XY	Un-defined	XY	XY	4
XY tolerance	.001	.001	0	.01	.001	5
Z tolerance	.001	.001	0	.01	.001	5
Active	False	F	True	F	F	6

---

**NOTE**

- 1 EPSG:4326 is a Layer Coordinate Reference System (CRS) as defined in the OpenGIS Web Map Service (WMS) Implementation Specification, OGC 04-024, version 1.3, dated August 2, 2004.
  - 2 A static extent type means that the extent is defined at the time of the creation of the spatial context using the `FdoICreateSpatialContext` command. A dynamic extent type means that the extent may change as features are added so that it always encloses all of the features.
  - 3 The default extents expressed in AGF text format for the various providers are as follows:
    - MySQL, : POLYGON ((-2000000 -2000000, 2000000 -2000000, 2000000 2000000, -2000000 2000000, -2000000 -2000000))
  - 4 The dimensionality is that of the extent. It is not set directly.
  - 5 See [Tolerance](#) (page 21).
  - 6 ArcSDE and MySQL providers can have multiple spatial contexts defined and can, therefore, show contexts in the reader that are not active.
-

## Exception Messages

### Connection Not Established

This exception message occurs when there is a closed connection to MySQL and a call is made to `EnumeratePropertyValues` on the connection properties dictionary object for the `DataStore` property.

### Incorrect usage of DB GRANT and GLOBAL PRIVILEGES

This exception message occurs when there is a pending connection to MySQL and a call is made to `GrantAccessToDatastore(<user>, <datastore>, <priv>)` on the user manage object where the privilege is not one of the defaults assigned by the call to `GrantAccessToDatastore(<user>, <datastore>)`. The privileges causing this exception are create user, file, process, reload, replication client, replication slave, show databases, shutdown, and super. This set is the difference between the list of privileges returned by `GetRolesOrPrivileges` and the defaults assigned by `GrantAccessToDatastore(<user>, <datastore>)`.





# 5

## **Installing RDBMS Clients Used By The Open Source Providers**

### **Introduction**

The operation of the ArcSDE provider requires the installation of an ArcSDE client on the same machine. This chapter describes the installation of an ArcSDE 9.1 client on Windows XP and Linux x86.

The operation of the MySQL provider requires the installation of an MySQL client on the same machine. This chapter describes the installation of a MySQL 5.0 database (server and client) on Windows XP and Linux x86.

The operation of the ODBC provider on Linux x86 requires the installation of an ODBC driver manager as well as ODBC provider-specific drivers. No installation is required on XP because XP is installed with the requisite software. This chapter describes the installation of the unixODBC driver manager and the Easysoft ODBC Oracle driver.

# Installing and Configuring an ArcSDE Client on Windows

## Installation

You must purchase the ArcSDE 9.1 client SDK from an ESRI vendor. There are instructions on how to purchase the client at <http://www.esri.com/software/arcgis/arcade/how-to-buy.html>.

Follow the instructions that come with the SDK to install the client to a folder outside of the FDO distribution. Make sure that the folder containing the following DLLs is visible in the %PATH% environment variable .

- pe91.dll
- sde91.dll
- sg91.dll

## Configuration

Before you can complete the configuration of the client on the local machine, you must install and configure the ArcSDE server to which the client will connect. An ArcSDE client does not have to be the same version as the ArcSDE server. Connecting an ArcSDE 9.0 or 9.1 client to an ArcSDE 8.3 server has been tested. Not all possible configurations are known.

It is possible for an ArcSDE client to connect directly to an Oracle or SQL Server database provided the required drivers are installed. This type of 'direct' connection has not been tested with the ArcSDE Provider, and so is not supported.

An ArcSDE server connects to only one kind of RDBMS. The ArcSDE Server for Oracle can connect to an Oracle database and the ArcSDE Server for SQL Server can connect to a Sql Server database. There are post-installations actions that you must take to create ArcSDE-specific tables in the target RDBMS before you can connect to the "ArcSDE-aware" database.

When you install an ArcSDE server, you configure it to listen on a port for database connect requests. Each ArcSDE server listens on its own dedicated listener port.

The ArcSDE client needs to know the number of the server's listener port and the protocol to use for sending requests to that port. The listener port and protocol is what the ArcSDE client needs to get from the ArcSDE provider in order to connect to the ArcSDE server. The ArcSDE provider API requires that the user put this information in the Instance connection property. The user can provide the information either directly or indirectly.

The direct way is simply the listener port and protocol. The syntax is "<port>/<protocol>", for example, "5151/tcp".

The indirect way is a symbolic name, for example, "esri\_sde\_oracle." The ArcSDE provider translates the symbolic name into a port and protocol by looking the symbolic name up in the `C:\WINDOWS\System32\drivers\etc\services` file. An example of an entry in the `services` file is `esri_sde_oracle 5151/tcp`.

You must also create usernames and passwords in each of the databases that you plan to connect to. You or your database administrator must connect to the database using an account with sufficient privileges to create a new user account and to grant the appropriate privileges to the new user.

## Installing and Configuring an ArcSDE Client on Linux

### Installation

You must purchase the ArcSDE 9.1 client SDK from an ESRI vendor. There are instructions on how to purchase the client at <http://www.esri.com/software/arcgis/arcsde/how-to-buy.html>.

Follow the instructions that come with the SDK to install the client to a directory outside of the Fdo distribution. Make sure that the folder containing the following shared objects is visible in the `LD_LIBRARY_PATH` environment variable .

- `libpe91.so`
- `libsde91.so`
- `libsg91.so`

## Configuration

Configuration on Linux is the same as configuration on Windows except that the location of the *services* file is */etc*. You need root privileges to modify this file.

## Installing and Configuring the MySQL Client on Windows

### Installation

Install the MySQL database, which includes the client as well as the server.

Download a Windows (x86) installer of the latest production release of MySQL 5.0 from <http://dev.mysql.com/downloads/mysql/5.0.html>. Do a 'typical' install to the default location (C:\Program Files\MySQL\MySQL Server 5.0).

The install wizard does not configure MySQL. At the conclusion of the installation procedure, you are given the option of continuing with configuration or leaving that step to later. If you decide to continue, you simply click the Finish button, which triggers the launch of the configuration wizard. If you decide to configure later, you invoke the MySQL Server Instance Config Wizard by following the menu path *Start > Programs > MySQL > MySQL Server 5.0*.

### Configuration

The configuration wizard presents you with the following configuration defaults. Go with the defaults except where noted otherwise.

- 1 Detailed Configuration
- 2 Multifunctional Database
- 3 default InnoDB Tablespace Setting
- 4 Decision Support (DSS)/OLAPA
- 5 Enable TCP/IP Networking (using Port Number 3306)
- 6 Enable Strict Mode

- 7 Use UTF-8 instead of the Standard Character Set
- 8 Install As Windows Service (Service Name = MySQL with the service started automatically)
- 9 Modify Security Settings (you must provide a root password (twice) and if you deinstalled an existing MySQL database, you must provide a current password)

You may optionally request that the *bin* directory be added the the Windows PATH environment variable. Do so. The wizard will add that directory to the end of %PATH%. The *bin* folder contains the *libmysql.dll* used by the MySQL provider.

---

**NOTE** If you have PHP installed on your machine, make sure that *C:\Program Files\MySQL\MySQL Server 5.0\bin* precedes the PHP path in the PATH environment variable. The PHP installation contains a copy of the *libmysql.dll* and it is important for the operation of the MySQL provider that the correct DLL is used.

---

The configuration wizard creates a configuration file (*C:\Program Files\MySQL\MySQL Server 5.0\my.ini*). It starts the MySQL service (mysqld) using options specified in *my.ini*.

You can use the Fdo User Manager API to create a user account with the usage privilege if you are using the enterprise version of FDO. Otherwise you can use the 2 SQL commands, `create user <user> identified by '<password>'` and `grant select, insert, update, delete, create, create view, drop, index, alter on *.* to '<user>'@'%' identified by '<password>'`. In either case you must be connected as a user, for example, root, with the grant privilege. The value <user> can be '<username>'@'localhost' if the connection local, or it must be '<username>'@',hostname.' if the connection is remote. The value of <hostname> is <host label>.<domain name>, for example, *someComputer.someCompany.com*.

---

**NOTE** If you enabled remote root access when you installed MySQL, you can create user accounts by way of a remote connection. Otherwise, you will have to connect locally in order to create a user account.

---

You can use the Fdo Data Store Management API to create a data store.

# Installing and Configuring the MySQL on Linux

## Installation

Install the MySQL database, which includes the client as well as the server.

Download a Linux x86 non RPM package Standard version (gzipped tarfile) containing the latest production release of MySQL 5.0 from <http://dev.mysql.com/downloads/mysql/5.0.html>. At the time of the writing of this document the latest version is 5.0.22 compiled with glibc-2.2.

Follow the installation instructions in the reference manual accessible at this URL <http://dev.mysql.com/doc/refman/5.0/en/index.html>. Click on 2. Installing and Upgrading MySQL. Click on 2.8 Installing MySQL on Other Unix-Like Systems.

---

**NOTE** The sequence of shell commands specified at the beginning of this section are executed with root privileges.

---

## Configuration

Configuration consists of setting up the mysql server as a service and of setting passwords for the 2 root and 2 anonymous accounts.

Return to the page containing the table of contents for Chapter 2. Click on 2. Installing and Upgrading MySQL. Click on 2.10.2 Unix Post-Installation Procedures. Scroll down to the paragraph whose first sentence is If you install MySQL from a source distribution or using a binary distribution format that does not install mysql.server automatically, you can install it manually. and read to the line containing the shell command `shell> chkconfig --add mysql`.

---

**NOTE** `chkconfig` uses a commented line from the `/etc/init.d/mysqlfile`, namely,  
`# chkconfig: 2345 64 36`

---

Verify that `chkconfig` has written the correct entries to the `/etc/rc*.d` directories.

- `/etc/rc0.d`, `/etc/rc1.d`, and `/etc/rc6.d` should contain `K36mysql -> ../init.d/mysql`
- `/etc/rc2.d`, `/etc/rc3.d`, `/etc/rc4.d`, and `/etc/rc5.d` should contain `S64mysql -> ../init.d/mysql`

Return to the page containing the table of contents for Chapter 2. Click on 2. Installing and Upgrading MySQL. Click on 2.10.3 Securing the Initial MySQL Accounts. Follow the instructions for securing the 2 root and 2 anonymous accounts.

---

**NOTE** If you get an error message like ERROR 2003 (HY00): Can't connect to MySQL server on '<hostname>' (<somenumber>) when setting the password for the '@<hostname>' or the 'root'@<hostname>' account, verify that the IP address that <hostname> maps to in the */etc/hosts* file is correct.

---

You can use the Fdo User Manager API to create a user account with the usage privilege if you are using the enterprise version of FDO. Otherwise you can use the 2 SQL commands, `create user <user> identified by '<password>'` and `grant select, insert, update, delete, create, create view, drop, index, alter on *.* to '<user>'@'%' identified by '<password>'`. In either case you must be connected as a user, for example, root, with the grant privilege. The value <user> can be '<username>'@'localhost' if the connection is local, or it must be '<username>'@'hostname.' if the connection is remote. The value of <hostname> is <host label>.<domain name>, for example, *someComputer.someCompany.com*.

---

**NOTE** If you enabled remote root access when you installed MySQL, you can create user accounts by way of a remote connection. Otherwise, you will have to connect locally in order to create a user account.

---

You can use the FDO Data Store Management API to create a data store.

## Installing and Configuring ODBC on Linux

First install the unixODBC driver manager and then install the drivers. Initially only the Easysoft ODBC Oracle driver is supported.

### Installation of the Driver Manager

Download the latest version of the driver manager from <http://www.unixODBC.org>. It is packaged as a gzipped tarfile. The file used for this exercise is *unixODBC-2.2.11.tar.gz*. It gunzips to a directory called *unixODBC-2.2.11*. Change directories to *unixODBC-2.2.11*, untar the file and run the following commands.

```
1 ./configure --prefix=/usr/local/unixODBC
```

- 2 `make`
- 3 `make install as root`



# Index

## A

API 17–18, 21, 30, 35, 79  
    capabilities 17  
    spatial context 21  
    user management 18, 30, 35, 79  
ArcSDE client 25, 72–74, 99–101  
    Installing on Linux 101  
    Installing on Windows 100  
ArcSDE server 72, 79, 100

## C

Capabilities API 17  
capabilities, command 18  
connection properties 21, 28, 44, 76,  
    79, 97

## D

data store 2, 19, 21, 34, 43, 90, 94  
Data Store Management API 19, 31  
data store, create or destroy 19, 21  
documentation 13

## E

Easysoft ODBC Oracle driver 99

## F

FDO provider 1, 5  
FDO Provider for ArcSDE 5, 21, 76, 79,  
    96, 99  
FDO Provider for MySQL 5, 19–21, 38,  
    77, 92, 94, 96–97, 99  
FDO Provider for ODBC 5, 77, 99  
FDO Provider for Oracle 6, 18–21, 38,  
    41, 43–44  
FDO Provider for Raster 6, 29, 44

FDO Provider for SDF 5, 19, 21, 77, 80  
FDO Provider for SHP 5, 77  
FDO Provider for SQL Server 6, 19–21,  
    30, 38, 42, 44–45  
FDO Provider for WFS 4, 6, 78  
FDO Provider for WMS 6, 78  
FdoIActivateSpatialContext 21  
FdoICommandCapabilities 19  
FdoICreateDataStore 19, 21  
FdoICreateSpatialContext 21, 44, 96  
FdoIDestroyDataStore 19, 21  
FdoIDestroySpatialContext 21  
FdoIGetSpatialContexts 21  
FdoIListDataStores 19, 21  
FdoUserManager 18, 35, 39, 93

## M

memory management 25–26, 73  
Method 36–41, 45, 91–93, 97  
    AddUser 40, 93  
    GetDbUsers 36, 39, 41, 91, 93  
    GetRolesOrPrivileges 37, 92, 97  
    GrantAccessToDatastore 37, 39–41,  
        45, 92–93, 97  
    RevokeAccessToDatastore 38, 41,  
        92  
    RevokeRoleOrPrivilege 38, 92  
    SupportsDataStoreGrants 38  
    SupportsWindowsAuthentication 38  
Microsoft Access 72, 80, 84–87  
Microsoft Excel 72, 80  
MySQL client 99, 102, 104  
    Installing on Linux 104  
    Installing on Windows 102  
MySQL database 39, 73–74, 79, 93, 99

## O

ODBC 105  
    Installing on Linux 105

ODBC data source 30, 83–87  
Open Source Geospatial Foundation 5  
Oracle 23–24, 71  
Oracle database 25–26, 79  
Oracle user 39

## P

privileges, MySQL 41, 94, 97  
privileges, SQL Server 39–41, 79, 93–94  
Property 29–31, 40, 42–44, 76–80, 94, 97  
ConnectionString 77  
DataSourceName 77  
Datastore 76  
DataStore 29, 31, 40, 42, 44, 77, 79, 94, 97  
DefaultFileLocation 77, 80  
DefaultImageHeight 78  
DefaultRasterFileLocation 29  
Description 42, 94  
FeatureServer 78  
File 77  
GenerateDefaultGeometryProperty 77  
Instance 76, 79  
LockMode 42–43  
LtMode 42–43  
Password 29–30, 42, 76–78  
ReadOnly 77  
Server 76, 79

Service 29–30, 77, 79  
TableSpace 42–43  
TemporaryFileLocation 78, 80  
UserId 77  
Username 29–30, 76–78

## R

Raster 24, 72  
RDBMS 5, 19, 23, 71  
roles, Oracle 39, 41, 93–94

## S

Spatial Context API 21  
spatial context properties 43, 94  
SQL command 19  
SQL Server 24, 72  
SQL Server database 23, 30–31, 79  
SQL Server user 39

## U

User Management API 18, 30, 35, 79

## W

Web Feature Service 72  
Web Map Service 72, 96