



ODBC Drivers for MobiLink and Remote Data Access

Part number: DC30129-01-0902-01
Last modified: October 2004

Copyright © 1989–2004 Sybase, Inc. Portions copyright © 2001–2004 iAnywhere Solutions, Inc. All rights reserved.

No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Sybase, Inc.

Sybase, SYBASE (logo), AccelaTrade, ADA Workbench, Adaptable Windowing Environment, Adaptive Component Architecture, Adaptive Server, Adaptive Server Anywhere, Adaptive Server Enterprise, Adaptive Server Enterprise Monitor, Adaptive Server Enterprise Replication, Adaptive Server Everywhere, Adaptive Server IQ, Adaptive Warehouse, AnswerBase, Anywhere Studio, Application Manager, AppModeler, APT Workbench, APT-Build, APT-Edit, APT-Execute, APT-Library, APT-Translator, ASEP, AvantGo, AvantGo Application Alerts, AvantGo Mobile Delivery, AvantGo Mobile Document Viewer, AvantGo Mobile Inspection, AvantGo Mobile Marketing Channel, AvantGo Mobile Pharma, AvantGo Mobile Sales, AvantGo Pylon, AvantGo Pylon Application Server, AvantGo Pylon Conduit, AvantGo Pylon PIM Server, AvantGo Pylon Pro, Backup Server, BayCam, Bit-Wise, BizTracker, Certified PowerBuilder Developer, Certified SYBASE Professional, Certified SYBASE Professional Logo, ClearConnect, Client Services, Client-Library, CodeBank, Column Design, ComponentPack, Connection Manager, Convoy/DM, Copernicus, CSP, Data Pipeline, Data Workbench, DataArchitect, Database Analyzer, DataExpress, DataServer, DataWindow, DB-Library, dbQueue, Developers Workbench, Direct Connect Anywhere, DirectConnect, Distribution Director, Dynamic Mobility Model, Dynamo, e-ADK, E-Anywhere, e-Biz Integrator, EC Gateway, ECMAP, ECRTP, eFulfillment Accelerator, Electronic Case Management, Embedded SQL, EMS, Enterprise Application Studio, Enterprise Client/Server, Enterprise Connect, Enterprise Data Studio, Enterprise Manager, Enterprise Portal (logo), Enterprise SQL Server Manager, Enterprise Work Architecture, Enterprise Work Designer, Enterprise Work Modeler, eProcurement Accelerator, eremote, Everything Works Better When Everything Works Together, EWA, E-Whatever, Financial Fusion, Financial Fusion (and design), Financial Fusion Server, Formula One, Fusion Powered e-Finance, Fusion Powered Financial Destinations, Fusion Powered STP, Gateway Manager, GeoPoint, GlobalFIX, iAnywhere, iAnywhere Solutions, ImpactNow, Industry Warehouse Studio, InfoMaker, Information Anywhere, Information Everywhere, InformationConnect, InstaHelp, Intelligent Self-Care, InternetBuilder, iremote, iScript, Jaguar CTS, jConnect for JDBC, KnowledgeBase, Logical Memory Manager, Mail Anywhere Studio, MainframeConnect, Maintenance Express, Manage Anywhere Studio, MAP, M-Business Channel, M-Business Network, M-Business Server, MDI Access Server, MDI Database Gateway, media.splash, Message Anywhere Server, MetaWorks, MethodSet, ML Query, MobiCATS, My AvantGo, My AvantGo Media Channel, My AvantGo Mobile Marketing, MySupport, Net-Gateway, Net-Library, New Era of Networks, Next Generation Learning, Next Generation Learning Studio, O DEVICE, OASiS, OASiS logo, ObjectConnect, ObjectCycle, OmniConnect, OmniSQL Access Module, OmniSQL Toolkit, Open Biz, Open Business Interchange, Open Client, Open Client/Server, Open Client/Server Interfaces, Open ClientConnect, Open Gateway, Open Server, Open ServerConnect, Open Solutions, Optima++, Orchestration Studio, Partnerships that Work, PB-Gen, PC APT Execute, PC DB-Net, PC Net Library, PhysicalArchitect, Pocket PowerBuilder, PocketBuilder, Power Through Knowledge, power.stop, Power++, PowerAMC, PowerBuilder, PowerBuilder Foundation Class Library, PowerDesigner, PowerDimensions, PowerDynamo, Powering the New Economy, PowerJ, PowerScript, PowerSite, PowerSocket, Powersoft, Powersoft Portfolio, Powersoft Professional, PowerStage, PowerStudio, PowerTips, PowerWare Desktop, PowerWare Enterprise, ProcessAnalyst, QAnywhere, Rapport, Relational Beans, RepConnector, Replication Agent, Replication Driver, Replication Server, Replication Server Manager, Replication Toolkit, Report Workbench, Report-Execute, Resource Manager, RW-DisplayLib, RW-Library, S.W.I.F.T. Message Format Libraries, SAFE, SAFE/PRO, SDF, Secure SQL Server, Secure SQL Toolset, Security Guardian, SKILLS, smart.partners, smart.parts, smart.script, SQL Advantage, SQL Anywhere, SQL Anywhere Studio, SQL Code Checker, SQL Debug, SQL Edit, SQL Edit/TPU, SQL Everywhere, SQL Modeler, SQL Remote, SQL Server, SQL Server Manager, SQL Server SNMP SubAgent, SQL Server/CFT, SQL Server/DBM, SQL SMART, SQL Station, SQL Toolset, SQLJ, Stage III Engineering, Startup.Com, STEP, SupportNow, Sybase Central, Sybase Client/Server Interfaces, Sybase Development Framework, Sybase Financial Server, Sybase Gateways, Sybase Learning Connection, Sybase MPP, Sybase SQL Desktop, Sybase SQL Lifecycle, Sybase SQL Workgroup, Sybase Synergy Program, Sybase User Workbench, Sybase Virtual Server Architecture, SybaseWare, Syber Financial, SyberAssist, SybMD, SyBooks, System 10, System 11, System XI (logo), SystemTools, Tabular Data Stream, The Enterprise Client/Server Company, The Extensible Software Platform, The Future Is Wide Open, The Learning Connection, The Model For Client/Server Solutions, The Online Information Center, The Power of One, TotalFix, TradeForce, Transact-SQL, Translation Toolkit, Turning Imagination Into Reality, UltraLite, UltraLite.NET, UNIBOM, Unilib, Uninull, Unisep, Unistring, URK Runtime Kit for UniCode, Versacore, Viewer, VisualWriter, VQL, Warehouse Control Center, Warehouse Studio, Warehouse WORKS, WarehouseArchitect, Watcom, Watcom SQL, Watcom SQL Server, Web Deployment Kit, Web.PB, Web.SQL, WebSights, WebViewer, WorkGroup SQL Server, XA-Library, XA-Server, and XP Server are trademarks of Sybase, Inc. or its subsidiaries.

All other trademarks are property of their respective owners.

Contents

About This Manual	v
SQL Anywhere Studio documentation	vi
Finding out more and providing feedback	ix
1 Introduction to iAnywhere Solutions ODBC Drivers	1
About iAnywhere Solutions ODBC drivers	2
Quick start for Windows	3
Quick start for UNIX	4
Error messages	9
2 iAnywhere Solutions ODBC Driver for Sybase Adaptive Server Enterprise	11
Introduction	12
Key settings for MobiLink and Adaptive Server Enterprise	13
Configuring data sources	15
Connecting to a data source using a logon dialog box	17
Connecting to a data source using a connection string	18
Data types	27
Unicode support	28
Default Unicode mapping	29
Supported features	30
3 iAnywhere Solutions ODBC Driver for Oracle	31
Introduction	32
Key settings for MobiLink and Oracle	33
Configuring data sources	34
Connecting to a Data Source Using a Logon Dialog Box	37
Connecting to a data source using a connection string	38
Data types	45
Unicode support	46
Default Unicode Mapping	47
Stored procedure results	48
Supported features	49
4 iAnywhere Solutions ODBC Driver for DB2	51
Introduction	52
Key settings for MobiLink and DB2	53
Configuring data sources	54

Connecting to a data source using a logon dialog box	59
Connecting to a data source using a connection string	60
Data types	65
Supported features	66
Unicode support	67
Default Unicode mapping	68
A ODBC Functions	69
ODBC functions	70
B Values for IANAAppCodePage Connection String Attribute	77
Table of IANAAppCodePage values	78
Index	83

About This Manual

Subject This manual describes how to use the iAnywhere Solutions ODBC drivers for Oracle, DB2, and Sybase Adaptive Server Enterprise. These ODBC drivers are included with SQL Anywhere Studio for use with MobiLink synchronization and Adaptive Server Anywhere remote data access.

Note

The ODBC driver for Adaptive Server Anywhere is described in the main Adaptive Server Anywhere documentation, and not in this book. For more information, see [“Working with ODBC data sources” \[ASA Database Administration Guide, page 53\]](#).

Audience This manual is for MobiLink synchronization users working with Oracle, DB2, or Sybase Adaptive Server Enterprise consolidated databases.

SQL Anywhere Studio documentation

The SQL Anywhere Studio documentation

This book is part of the SQL Anywhere documentation set. This section describes the books in the documentation set and how you can use them.

The SQL Anywhere Studio documentation is available in a variety of forms: in an online form that combines all books in one large help file; as separate PDF files for each book; and as printed books that you can purchase. The documentation consists of the following books:

- ◆ **Introducing SQL Anywhere Studio** This book provides an overview of the SQL Anywhere Studio database management and synchronization technologies. It includes tutorials to introduce you to each of the pieces that make up SQL Anywhere Studio.
- ◆ **What's New in SQL Anywhere Studio** This book is for users of previous versions of the software. It lists new features in this and previous releases of the product and describes upgrade procedures.
- ◆ **Adaptive Server Anywhere Database Administration Guide** This book covers material related to running, managing, and configuring databases and database servers.
- ◆ **Adaptive Server Anywhere SQL User's Guide** This book describes how to design and create databases; how to import, export, and modify data; how to retrieve data; and how to build stored procedures and triggers.
- ◆ **Adaptive Server Anywhere SQL Reference Manual** This book provides a complete reference for the SQL language used by Adaptive Server Anywhere. It also describes the Adaptive Server Anywhere system tables and procedures.
- ◆ **Adaptive Server Anywhere Programming Guide** This book describes how to build and deploy database applications using the C, C++, and Java programming languages. Users of tools such as Visual Basic and PowerBuilder can use the programming interfaces provided by those tools. It also describes the Adaptive Server Anywhere ADO.NET data provider.
- ◆ **Adaptive Server Anywhere SNMP Extension Agent User's Guide** This book describes how to configure the Adaptive Server Anywhere SNMP Extension Agent for use with SNMP management applications to manage Adaptive Server Anywhere databases.
- ◆ **Adaptive Server Anywhere Error Messages** This book provides a complete listing of Adaptive Server Anywhere error messages together with diagnostic information.

-
- ◆ **SQL Anywhere Studio Security Guide** This book provides information about security features in Adaptive Server Anywhere databases. Adaptive Server Anywhere 7.0 was awarded a TCSEC (Trusted Computer System Evaluation Criteria) C2 security rating from the U.S. Government. This book may be of interest to those who wish to run the current version of Adaptive Server Anywhere in a manner equivalent to the C2-certified environment.
 - ◆ **MobiLink Administration Guide** This book describes how to use the MobiLink data synchronization system for mobile computing, which enables sharing of data between a single Oracle, Sybase, Microsoft or IBM database and many Adaptive Server Anywhere or UltraLite databases.
 - ◆ **MobiLink Clients** This book describes how to set up and synchronize Adaptive Server Anywhere and UltraLite remote databases.
 - ◆ **MobiLink Tutorials** This book provides several tutorials that walk you through how to set up and run MobiLink applications.
 - ◆ **MobiLink Server-Initiated Synchronization User's Guide** This book describes MobiLink server-initiated synchronization, a feature of MobiLink that allows you to initiate synchronization from the consolidated database.
 - ◆ **QAnywhere User's Guide** This manual describes MobiLink QAnywhere, a messaging platform that enables the development and deployment of messaging applications for mobile and wireless clients, as well as traditional desktop and laptop clients.
 - ◆ **iAnywhere Solutions ODBC Drivers** This book describes how to set up ODBC drivers to access consolidated databases other than Adaptive Server Anywhere from the MobiLink synchronization server and from Adaptive Server Anywhere remote data access.
 - ◆ **SQL Remote User's Guide** This book describes all aspects of the SQL Remote data replication system for mobile computing, which enables sharing of data between a single Adaptive Server Anywhere or Adaptive Server Enterprise database and many Adaptive Server Anywhere databases using an indirect link such as e-mail or file transfer.
 - ◆ **SQL Anywhere Studio Help** This book includes the context-sensitive help for Sybase Central, Interactive SQL, and other graphical tools. It is not included in the printed documentation set.
 - ◆ **UltraLite Database User's Guide** This book is intended for all UltraLite developers. It introduces the UltraLite database system and provides information common to all UltraLite programming interfaces.

-
- ◆ **UltraLite Interface Guides** A separate book is provided for each UltraLite programming interface. Some of these interfaces are provided as UltraLite components for rapid application development, and others are provided as static interfaces for C, C++, and Java development.

In addition to this documentation set, PowerDesigner and InfoMaker include their own online documentation.

Documentation formats SQL Anywhere Studio provides documentation in the following formats:

- ◆ **Online documentation** The online documentation contains the complete SQL Anywhere Studio documentation, including both the books and the context-sensitive help for SQL Anywhere tools. The online documentation is updated with each maintenance release of the product, and is the most complete and up-to-date source of documentation.

To access the online documentation on Windows operating systems, choose Start ► Programs ► SQL Anywhere 9 ► Online Books. You can navigate the online documentation using the HTML Help table of contents, index, and search facility in the left pane, as well as using the links and menus in the right pane.

To access the online documentation on UNIX operating systems, see the HTML documentation under your SQL Anywhere installation.

- ◆ **PDF books** The SQL Anywhere books are provided as a set of PDF files, viewable with Adobe Acrobat Reader.

The PDF books are accessible from the online books, or from the Windows Start menu.

- ◆ **Printed books** The complete set of books is available from Sybase sales or from eShop, the Sybase online store, at <http://eshop.sybase.com/eshop/documentation>.

Finding out more and providing feedback

Finding out more

Additional information and resources, including a code exchange, are available at the iAnywhere Developer Network at <http://www.ianywhere.com/developer/>.

If you have questions or need help, you can post messages to the iAnywhere Solutions newsgroups listed below.

When you write to one of these newsgroups, always provide detailed information about your problem, including the build number of your version of SQL Anywhere Studio. You can find this information by typing **dbeng9 -v** at a command prompt.

The newsgroups are located on the *forums.sybase.com* news server. The newsgroups include the following:

- ◆ [sybase.public.sqlanywhere.general](#)
- ◆ [sybase.public.sqlanywhere.linux](#)
- ◆ [sybase.public.sqlanywhere.mobilink](#)
- ◆ [sybase.public.sqlanywhere.product_futures_discussion](#)
- ◆ [sybase.public.sqlanywhere.replication](#)
- ◆ [sybase.public.sqlanywhere.ultralite](#)
- ◆ [ianywhere.public.sqlanywhere.qanywhere](#)

Newsgroup disclaimer

iAnywhere Solutions has no obligation to provide solutions, information or ideas on its newsgroups, nor is iAnywhere Solutions obliged to provide anything other than a systems operator to monitor the service and ensure its operation and availability.

iAnywhere Solutions Technical Advisors as well as other staff assist on the newsgroup service when they have time available. They offer their help on a volunteer basis and may not be available on a regular basis to provide solutions and information. Their ability to help is based on their workload.

Feedback

We would like to receive your opinions, suggestions, and feedback on this documentation.

You can e-mail comments and suggestions to the SQL Anywhere documentation team at iasdoc@ianywhere.com. Although we do not reply to e-mails sent to that address, we read all suggestions with interest.

In addition, you can provide feedback on the documentation and the software through the newsgroups listed above.

CHAPTER 1

Introduction to iAnywhere Solutions ODBC Drivers

About this chapter

This chapter introduces the iAnywhere Solutions ODBC drivers that are included in SQL Anywhere Studio for use with MobiLink synchronization or Adaptive Server Anywhere remote data access. It provides general information and key settings. For details of how to configure the iAnywhere Solutions ODBC drivers, see:

- ◆ “iAnywhere Solutions ODBC Driver for Sybase Adaptive Server Enterprise” on page 11
- ◆ “iAnywhere Solutions ODBC Driver for Oracle” on page 31
- ◆ “iAnywhere Solutions ODBC Driver for DB2” on page 51

Note

The ODBC driver for Adaptive Server Anywhere is described in the main Adaptive Server Anywhere documentation, and not in this book. For more information, see “Working with ODBC data sources” [*ASA Database Administration Guide*, page 53].

Contents

Topic:	page
About iAnywhere Solutions ODBC drivers	2
Quick start for Windows	3
Quick start for UNIX	4
Error messages	9

About iAnywhere Solutions ODBC drivers

iAnywhere Solutions ODBC drivers are compliant with the Open Database Connectivity (ODBC) specification. ODBC is a specification for an application program interface (API) that enables applications to access multiple database management systems using SQL.

The iAnywhere Solutions ODBC drivers are for use only with MobiLink synchronization server and administration tools, and with Adaptive Server Anywhere remote data access. Other ODBC client applications cannot use these drivers.

iAnywhere Solutions ODBC drivers are provided for the following operating systems:

- ◆ Windows operating systems, except Windows CE and 64-bit drivers
- ◆ UNIX operating systems: Solaris, AIX, and Linux

The iAnywhere Solutions ODBC drivers are installed by the SQL Anywhere Studio Setup program.

Note

The ODBC driver for Adaptive Server Anywhere is described in the main Adaptive Server Anywhere documentation, and not in this book. For more information, see [“Working with ODBC data sources” \[ASA Database Administration Guide, page 53\]](#).

Quick start for Windows

The iAnywhere Solutions ODBC drivers support the following Windows operating systems:

- ◆ Windows 95, Windows 98, Windows Me, Windows XP
- ◆ Windows NT with Service Pack 5 and higher
- ◆ Windows 2000 with Service Pack 1 and higher

On Windows, the ODBC drivers are 32-bit drivers. All required network software supplied by your database system vendors must be 32-bit compliant.

Starting the ODBC Administrator

The ODBC Administrator is used to manage data sources created for each driver.

❖ To start the ODBC Administrator (Windows)

1. Double-click the ODBC Data Sources icon in the Control Panel.

❖ To start the ODBC Administrator (command line)

1. At a command prompt, enter the command `odbcad32`.

Once in the ODBC Administrator, you can create a new data source by clicking Add and choosing one of the iAnywhere Solutions ODBC drivers.

Driver file names

The prefix for all iAnywhere Solutions ODBC driver file names is *wq*. The file extension is *.dll*. This indicates that they are dynamic link libraries.

Quick start for UNIX

The following UNIX operating systems are supported:

- ◆ Solaris
- ◆ Linux (Red Hat, Caldera, and SuSE)
- ◆ AIX

☞ For more information about the versions that are supported, see the *UNIX Readme First*, which is installed in your SQL Anywhere installation directory in two formats: *readme.txt* and *readme.pdf*.

The system information file (.odbc.ini)

In the UNIX environment, there is no ODBC administrator. To configure a data source, you must edit the system information file, a plain text file that is normally located in your *\$HOME* directory and is usually called *.odbc.ini*. This file is maintained using any text editor, and defines data source entries as described in the following sections:

- ◆ **Sybase Adaptive Server Enterprise** “Connecting to a data source using a connection string” on page 18.
- ◆ **Oracle** “Connecting to a data source using a connection string” on page 38.
- ◆ **IBM DB2** “Connecting to a data source using a connection string” on page 60.

You must use the long name of connection string attributes when defining data source entries. A sample file (*odbc.ini.sample*) is located in the driver installation directory.

Caution

You should not add simple encryption to the .odbc.ini system information file with the File Hiding utility (dbfhide) on UNIX unless you will only be using Adaptive Server Anywhere data sources. If you plan to use other data sources (for example, for MobiLink synchronization), then obfuscating the contents of the .odbc.ini file may prevent other drivers from functioning properly.

☞ For more information, see “Hiding the contents of .ini files” [ASA Database Administration Guide, page 524].

System information file contents

There must be an [ODBC] section in the system information file that includes the InstallDir keyword. The value of this keyword must be the path

to the directory under which the `/lib` and `/messages` directories are contained. For example, if you choose the default install directory, then the following line must be in the [ODBC] section:

```
InstallDir=/opt/sybase/SYBSsa9/drivers
```

Sample system
information file

The following is a sample system information file for Solaris:

```
; This is only a sample .odbc.ini file
; IT WILL NOT WORK AS IS.
; You need to make a copy of this file
; into the user's home directory and modify
; the DSNs (Data Source Names) to your needs:
;
; cp odbc.ini.sample ~/.odbc.ini
; vi ~/.odbc.ini
;
; You also need to modify the default
; value of your ODBCINI environment variable.
; It should be set to point to your local
; copy of the .odbc.ini file:
; export ODBCINI=~/.odbc.ini
;::::::::::::::::::::::::::::::::::::::::::

[ODBC Data Sources]
SybaseASA=Adaptive Server Anywhere 9 Driver
Oracle Wire Protocol=iAnywhere Solutions 9 - Oracle Wire
    Protocol Driver
Sybase ASE Wire Protocol=iAnywhere Solutions 9 - Sybase ASE
    Driver
DB2 Wire Protocol=iAnywhere Solutions 9 - DB2 Wire Protocol
    Driver

[SybaseASA]
Driver=/opt/sybase/SYBSsa9/lib/libdbodbc9_r.so
UID=dba
PWD=sql
ServerName=asademo
CommLinks=tcPIP
```

```
[Oracle Wire Protocol]
Driver=/opt/sybase/SYBSsa9/drivers/lib/wqora19.so
Description=iAnywhere Solutions 9 - Oracle Wire Protocol
ApplicationUsingThreads=1
ArraySize=60000
CachedCursorLimit=32
CachedDescLimit=0
CatalogIncludesSynonyms=1
CatalogOptions=0
DefaultLongDataBuffLen=1024
DescribeAtPrepare=0
EnableDescribeParam=0
EnableNcharSupport=0
EnableScrollableCursors=1
EnableStaticCursorsForLongData=0
EnableTimestampWithTimeZone=0
HostName=oracleserver
LocalTimeZoneOffset=
LockTimeOut=-1
LogonID=uid
Password=pwd
PortNumber=1521
ProcedureRetResults=1
SID=oraclesid
UseCurrentSchema=1
```

```
[Sybase ASE Wire Protocol]
Driver=/opt/sybase/SYBSsa9/drivers/lib/wqase19.so
Description=iAnywhere Solutions 9 - Sybase ASE Wire Protocol
ApplicationName=
ApplicationUsingThreads=1
ArraySize=50
Charset=
CursorCacheSize=1
Database=db
DefaultLongDataBuffLen=1024
EnableDescribeParam=1
EnableQuotedIdentifiers=0
InitializationString=
Language=
LogonID=uid
NetworkAddress=serverhost,4100
OptimizePrepare=2
PacketSize=0
Password=pwd
RaiseErrorPositionBehavior=0
SelectMethod=0
WorkStationID=
```



```
[DB2 Wire Protocol]
Driver=/opt/sybase/SYBSsa9/drivers/lib/wqdb219.so
Description=iAnywhere Solutions 9 - DB2 Wire Protocol
AddStringToCreateTable=
AlternateID=
Collection=
Database=db
DynamicSections=100
GrantAuthid=PUBLIC
GrantExecute=1
IpAddress=db2host
IsolationLevel=CURSOR_STABILITY
Location=
LogonID=uid
Password=pwd
Package=pkg
PackageOwner=uid
TcpPort=50000
WithHold=1

[ODBC]
IANAAppCodePage=4
InstallDir=/opt/sybase/SYBSsa9/drivers
Trace=0
TraceDll=/opt/sybase/SYBSsa9/drivers/lib/odbc trac.so
TraceFile=odbc trace.out
UseCursorLib=0
```

Using a centralized file

UNIX support of the database drivers also permits the use of a centralized system information file that a system administrator can control. This is accomplished by setting the environment variable ODBCINI to point to the fully qualified path name of the centralized file. For example, in the C shell you could set this variable as follows:

```
setenv ODBCINI /opt/odbc/system_odbc.ini
```

In the Bourne or Korn shell, you would set it as:

```
ODBCINI=/opt/odbc/system_odbc.ini;export ODBCINI
```

The search order for the location of the system information file is as follows:

1. Check ODBCINI
2. Check $\$HOME$ for *.odbc.ini*

Driver file names

The iAnywhere Solutions ODBC drivers are ODBC API-compliant dynamic link libraries, referred to in UNIX as **shared objects**. The prefix for all ODBC driver file names on UNIX is *wq*. On UNIX, the driver file names are lowercase and the extension is *.so*.

Environment variables

To use iAnywhere Solutions ODBC drivers on UNIX, there are several environment variables that you may need to set. It is recommended that you use the appropriate file for your shell, either *asa_config.sh* or *asa_config.csh* (located in the directory */opt/sybase/SYBSsa9/bin*) as a template for setting the required environment variables. Some of the environment variables set by the *asa_config* files include `PATH`, `LD_LIBRARY_PATH`, `ASANY9`, and `ASANYSH9`.

Library Path environment variable You must include the full path to the dynamic link libraries in the environment variable `LD_LIBRARY_PATH`. For example, if you install the ODBC drivers in the system directory */opt/sybase/SYBSsa9/drivers*, then the fully qualified path for the ODBC pack is */opt/sybase/SYBSsa9/drivers/lib*.

If you do not include the path */opt/sybase/SYBSsa9/drivers/lib* in the environment variable `LD_LIBRARY_PATH`, then your applications are unable to load the ODBC drivers dynamically at runtime or to display error message text.

Additional environment variables Some iAnywhere Solutions ODBC drivers must have environment variables set as required by the database client components used by the drivers. Consult the driver requirements in each of the individual driver sections for additional information pertaining to individual driver requirements.

`ODBCINI` is an optional environment variable that all iAnywhere Solutions ODBC drivers recognize. `ODBCINI` is used to locate an ODBC information file other than the default file.

Linux Redhat version 9 To use the iAnywhere Solutions ODBC driver on Linux Redhat version 9 and Linux Redhat Enterprise Server 3.0, you must also set the following environment variable:

```
LD_ASSUME_KERNEL=2.4.1
```

Error messages

Error messages can come from the following sources:

- ◆ An ODBC driver
- ◆ The database system
- ◆ The ODBC driver manager

ODBC driver errors

An error reported on an ODBC driver has the following format:

```
[vendor] [ODBC_component] message
```

where *ODBC_component* is the component in which the error occurred. If you receive this type of error, check the last ODBC call made by your application for possible problems or contact your ODBC application vendor.

An error that occurs in the data source includes the data store name, in the following format:

```
[vendor] [ODBC_component] [data_store] message
```

With this type of message, *ODBC_component* is the component that received the error from the data store indicated. For example, you may receive the following message from an Oracle data store:

```
[DataDirect] [ODBC Oracle driver] [Oracle] ORA-0919: specified
length too long for CHAR column
```

If you receive this type of error, something is wrong with the database system. Check your database system documentation for more information or consult your database administrator. In this example, you would check your Oracle documentation.

Driver Manager errors

The driver manager is a DLL or shared object that establishes connections with drivers, submits requests to drivers, and returns results to applications. An error that occurs in the driver manager has the following format:

```
[vendor] [ODBC XXX] message
```

For example, an error from the Microsoft driver manager might look like this:

```
[Microsoft] [ODBC Driver Manager] Driver does not support this
function
```

If you receive this type of error, consult the Programmer's Reference for the Microsoft ODBC Software Development Kit that is available from Microsoft.

UNIX error handling

UNIX error handling follows the X/Open XPG3 messaging catalog system. Localized error messages are stored in the subdirectory *locale/localized_territory_directory/LC_MESSAGES*, where *localized_territory_directory* depends on your language.

For instance, German localization files are stored in *locale/de/LC_MESSAGES*, where *de* is the locale for German.

If localized error messages are not available for your locale, then they will contain message numbers instead of text. For example:

```
[DataDirect] [ODBC 20101 driver] 30040
```

CHAPTER 2

iAnywhere Solutions ODBC Driver for Sybase Adaptive Server Enterprise

About this chapter

This chapter describes how to configure and use the iAnywhere Solutions ODBC driver for Sybase Adaptive Server Enterprise databases.

Contents

Topic:	page
Introduction	12
Key settings for MobiLink and Adaptive Server Enterprise	13
Configuring data sources	15
Connecting to a data source using a logon dialog box	17
Connecting to a data source using a connection string	18
Data types	27
Unicode support	28
Default Unicode mapping	29
Supported features	30

Introduction

The iAnywhere Solutions ODBC driver for Sybase Adaptive Server Enterprise supports Adaptive Server Enterprise 11.0 and higher database systems in the Windows and UNIX environments.

Driver requirements

The Adaptive Server Enterprise driver is a wire protocol driver, which means that there is no additional client software required.

Key settings for MobiLink and Adaptive Server Enterprise

The ODBC driver for Adaptive Server Enterprise does not require any Adaptive Server Enterprise client software. The following are key settings for Adaptive Server Enterprise. Other parameters are available and described elsewhere, but are less important.

- ◆ **Basic connection settings** Set the following basic connection settings:
 - **Data source name** This value is used to identify the data source. On Windows, it is located on the General tab. On UNIX, this attribute is called DataSourceName (DSN).
 - **Network library name** For TCP/IP networks, set this to Winsock. On Windows, it is located on the General tab. On UNIX, this attribute is called NetworkLibraryName (NLM).
 - **Network address and port of the server (General tab)** These should be separated by commas; for example, **server-machine,1500**. On Windows, it is located on the General tab. On UNIX, this attribute is called NetworkAddress (NA).
 - **Default Logon ID used for the connection** This is case sensitive and must correspond to a valid logon ID on the server. On Windows, it is located on the Connection tab. On UNIX, this attribute is called LogonID (UID).
- ◆ **DefaultLongDataBuffLen** This parameter specifies (in Kb) the size of the largest LONG BINARY or LONG VARCHAR column value being synchronized. The default setting is 1024, which is equivalent to 1 Mb. If you are synchronizing long columns with entries larger than this, set the value higher.

This parameter appears on the Advanced tab of the Windows driver ODBC setup dialog. On UNIX, this attribute is called DefaultLongDataBuffLen (DLDBL).
- ◆ **SelectMethod** This parameter can be 0 or 1. When you use 0 (the default), database cursors are used. When set to 1, SELECT statements are run directly, without cursors.

This parameter appears on the Performance tab of the Windows driver ODBC setup dialog. On UNIX, this attribute is called SelectMethod (SM).
- ◆ **OptimizePrepare** This parameter can be 0, 1, 2, or 3. For use with MobiLink, set it to 2.

This parameter appears on the Performance tab of the Windows driver ODBC setup dialog. On UNIX, this attribute is called OptimizePrepare (OP).

☞ For complete information about settings for the Adaptive Server Enterprise driver, see [“Connecting to a data source using a connection string”](#) on page 18.

Configuring data sources

On Windows, data sources are configured and modified through the ODBC Administrator.

❖ To configure a Sybase Adaptive Server Enterprise data source (UNIX)

1. In the UNIX environment, there is no ODBC administrator. To configure a data source in the UNIX environment, you must edit the system information file.

☞ For information about editing this file, see [“Quick start for UNIX” on page 4](#).

❖ To configure a Sybase Adaptive Server Enterprise data source (Windows)

1. Start the ODBC Administrator to display a list of data sources.
2. If you are configuring an existing data source, select the data source name and click **Configure** to display the Adaptive Server Enterprise Driver Setup dialog box.
3. If you are configuring a new data source, click **Add** to display a list of installed drivers. Select the **iAnywhere Solutions 9 - Sybase ASE** driver and click **Finish** to display the Adaptive Server Enterprise Driver Setup dialog box.
4. On the General tab, provide the following connection information and then click Apply.
 - ◆ **Data Source Name** Type a string that identifies this Adaptive Server Enterprise data source configuration in the system information. Examples include **Accounting** or **Sys11-Serv1**.
 - ◆ **Description** Type an optional long description of a data source name. For example, **My Accounting Database** or **System 11 on Server number 1**.
 - ◆ **Network Library Name** Select the name of the network library. This specifies which network protocol to use. The values are Winsock and NamedPipes.
 - ◆ **Network Address** Type the network address. The value you specify depends on which network protocol is chosen under Network Library Name and on the Adaptive Server Enterprise server.
If you choose Winsock for the Network Library Name, specify an IP address as follows: *servername-or-IP-address, port-number*. For

example, if your network supports named servers, you may specify an address such as **Sybaseserver, 5000**. You may also specify the IP address directly such as **199.226.224.34, 5000**.

If you choose NamedPipes as the network protocol, you must specify the pipe address of the server. For example,
\\machine1\sybase\pipe\query.

- ◆ **Database Name** Type the name of the database to which you want to connect by default. If you do not specify a value, the default is the database defined by the system administrator for each user.
5. Configure other settings in a manner appropriate for your application.
 - ☞ Most settings can be left at their default values. For a list of key driver settings when using MobiLink, see [“Key settings for MobiLink and Adaptive Server Enterprise” on page 13](#).
 6. At any point during the configuration process, you can click **Test Connect** to attempt to connect to the data source using the connection properties specified in the Driver Setup dialog box. A logon dialog box is displayed. For more information, see [“Connecting to a data source using a logon dialog box” on page 17](#). Note that the information you enter in the logon dialog box during a test connect is not saved.
 - ◆ If the driver can connect, it releases the connection and displays a “connection established” message. Click **OK**.
 - ◆ If the driver cannot connect because of an improper environment or incorrect connection value, it will display an appropriate error message. Click **OK**.
 7. Click **OK** or **Cancel**. If you click **OK**, the values you have specified become the defaults when you connect to the data source. You can change these defaults by using this procedure to reconfigure your data source. You can override these defaults by connecting to the data source using a connection string with alternate values.

Connecting to a data source using a logon dialog box

Some ODBC applications display a Logon dialog box when you are connecting to a data source. In these cases, the data source name has already been specified.

In this dialog box, perform the following steps:

1. In the Network Library field, select the name of the network library. This specifies which network protocol to use. Valid values are Winsock and NamedPipes.
2. In the Network Address field, type the network address, which depends on the network protocol chosen under Network Library and on the Adaptive Server Enterprise server. If you choose Winsock, specify an IP address as follows: *servername-or-IP-address, port-number*. For example, if your network supports named servers, you may specify an address such as **Sybaseserver, 5000**. You may also specify the IP address directly such as **199.226.224.34, 5000**.

If you have chosen NamedPipes as the network protocol, you must specify the pipe address of the server. For example,
`\\machine1\sybase\pipe\query.`

3. If required, type your case-sensitive login ID.
4. If required, type your case-sensitive password for the system.
5. In the Database field, type the name of the database you want to access (case-sensitive) or select the name from the Database drop-down list, which displays the names you specified in the Adaptive Server Enterprise Driver Setup dialog box.
6. Click **OK** to complete the logon and to update the values in the system information.

Connecting to a data source using a connection string

If your application requires a connection string to connect to a data source, you must specify the data source name that tells the driver which section in the system information to use for the default connection information. Optionally, you may specify attribute=value pairs in the connection string to override the default values stored in the system information. These values are not written to the system information.

You can specify either long or short names in the connection string. The connection string has the form:

```
DSN=data_source_name[;attribute=value[;attribute=value]...]
```

An example of a connection string for Adaptive Server Enterprise is:

```
DSN=SYS11 TABLES;SRVR=QESRVR;DB=PAYROLL;UID=JOHN;PWD=XYZZY
```

The following table gives the long and short names for each attribute, as well as a description. To configure a data source in the UNIX environment, you must edit the system information file. This file accepts only long names for attributes.

☞ For information about editing this file, see [“Quick start for UNIX” on page 4](#).

The defaults listed in the table are initial defaults that apply when no value is specified in either the connection string or in the data source definition in the system information. If you specified a value for the attribute when configuring the data source, that value is the default.

☞ Most settings can be left at their default values. For a list of key driver settings when using MobiLink, see [“Key settings for MobiLink and Adaptive Server Enterprise” on page 13](#).

Attribute	Description
AppCodePage (ACP)	AppCodePage has been replaced by IANAAppCodePage (IACP). See below.
ApplicationName (APP)	The name used by Sybase to identify your application.

Attribute	Description
ApplicationUsingThreads (AUT)	<p>ApplicationUsingThreads={0 1}. Ensures that the driver works with multi-threaded applications.</p> <p>When set to 1 (the initial default), the driver is thread-safe.</p> <p>When using the driver with single-threaded applications, you can set this option to 0 to avoid additional processing required for ODBC thread-safety standards.</p>
ArraySize (AS)	<p>The number of rows the driver retrieves from the server for a fetch. This is not the number of rows given to the user. This increases performance by reducing network traffic.</p> <p>The initial default is 50 rows.</p>
Charset (CS)	<p>The name of a character set. This character set must be installed on the Sybase server. The default is the setting on the Sybase server. For this driver to support Unicode, this attribute must be set to UTF-8. Refer to the Sybase server documentation for a list of valid character set names.</p>
CursorCacheSize (CCS)	<p>The number of connections that the connection cache can hold. To set the connection cache, you must set the SelectMethod attribute to 1. Increasing the connection cache may increase performance of some applications but requires additional database resources.</p> <p>The initial default is 1 (one cursor).</p>
Database (DB)	<p>The name of the database to which you want to connect.</p>
DataSourceName (DSN)	<p>A string that identifies a single connection to a Sybase database. Examples include "Accounting" or "Sys10-Serv1."</p>
DefaultLongDataBuffLen (DLDBL)	<p>An integer value that specifies, in 1024-byte multiples, the maximum length of data fetched from a TEXT or IMAGE column. You will need to increase this value if the total size of any long data exceeds 1 MB.</p> <p>The default is 1024.</p>

Attribute	Description
DistributedTransaction-Model (DTM)	<p>DistributedTransactionModel={XA Protocol 21 Native OLE}. Determines which model is used for distributed transaction support. The initial default is XA Protocol.</p> <p>Leave this option at the default setting.</p>
EnableDescribeParam (EDP)	<p>EnableDescribeParam={0 1}. Determines whether the ODBC API function SQLDescribeParam is enabled.</p> <p>When set to 0 (the initial default), SQLDescribeParam is disabled.</p> <p>When set to 1, SQLDescribeParam is enabled, which allows an application to describe parameters in SQL statements and in stored procedure calls. To use this option, OptimizePrepare must be set to 0 or 1, and the SQL statement must not include long parameters. This attribute should be set to 1 when using Microsoft Remote Data Objects (RDO) to access data.</p>
EnableQuotedIdentifiers (EQI)	<p>EnableQuotedIdentifiers={0 1}. Enables quoted identifiers.</p> <p>When set to 0 (the initial default), quoted identifiers are disabled.</p> <p>When set to 1, quoted identifiers are enabled.</p>
FailoverNetworkAddress (FNA)	<p>Specifies the address of the High Availability (HA) Failover server to be used in the event of a connection loss. The driver detects the dropped connection and automatically reconnects to the HA Failover server specified by this attribute. This attribute is valid only for Sybase version 12 or higher servers that have the High Availability Failover feature enabled.</p> <p>See the description of the Network Address attribute for an explanation of valid values.</p>

Attribute	Description
IANAAppCodePage (IACP)	<p>Valid values for this attribute are listed in “Values for IANAAppCodePage Connection String Attribute” on page 77. The code page that you specify must be the same as the code page used by your application. The driver on UNIX determines the value of the application’s code page by checking for an IANAAppCodePage value in the following order:</p> <ul style="list-style-type: none"> ◆ In the connection string ◆ In the DataSource section of the system file (odbc.ini) ◆ In the ODBC section of the system file (odbc.ini) <p>If no IANAAppCodePage value is found, the driver uses the default value of 4 (ISO 8859-1 Latin-1).</p> <p>NOTE: The IANAAppCodePage connection string attribute replaces the AppCodePage connection string attribute in earlier versions of Connect for ODBC. The drivers are backwardly compatible with the AppCodePage attribute, but you must now use the IANAAppCodePage attribute.</p>
InitializationString (IS)	<p>InitializationString={ Sybase set commands ; . . . }. Supports the execution of Sybase commands at connect time. Multiple commands must be separated by semicolons.</p>
InterfacesFile (IF)	<p>Specifies the path name of the Interfaces file. If you do not specify a value for this attribute, but specify a value for the Interfaces File Server Name attribute, the driver looks for the path name of the Interfaces file in the Registry under HKEY_LOCAL_MACHINE\SOFTWARE\DataDirect\InterfacesFile. If this Registry value is empty, then the driver attempts to open the SQL.INI file found in the same directory as the driver and use it as the Interfaces file.</p>

Attribute	Description
InterfacesFileServer Name (IFSN)	Specifies the name of the section in the Interfaces file that contains the network connection information for the Sybase server you want to access. The section name typically is the host name of the server that contains the Sybase Server you want to access.
Language (LANG)	<p>The national language. This language must be installed on the Sybase server.</p> <p>The initial default is English.</p>
LogonID (UID)	The default logon ID used to connect to your Sybase database. This ID is case-sensitive. A logon ID is required only if security is enabled on your database. If so, contact your system administrator to get your logon ID.
NetworkAddress (NA)	<p>The network address depends on which network protocol is chosen under Network Library Name and on the Sybase server. If you have chosen Winsock, specify an IP address as follows: <i>"servername-or-IP-address, port-number"</i>. For example, if your network supports named servers, you may specify an address such as "Sybaseserver, 5000". You may also specify the IP address directly such as "199.226.224.34, 5000".</p> <p>If you have chosen NamedPipes as the network protocol, you must specify the pipe address of the server. For example, <i>"\\machine1\sybase\pipe\query"</i>.</p>
NetworkLibraryName (NLM)	<p>NetworkLibraryName={Winsock NamedPipes}. The name of the network library. This specifies which network protocol to use.</p> <p>The initial default is Winsock.</p> <p>This option has no effect on UNIX; on UNIX, TCP/IP is used.</p>

Attribute	Description
OptimizePrepare (OP)	<p>OptimizePrepare={0 1 2 3}. Determines whether stored procedures are created on the server for calls to SQLPrepare.</p> <p>For use with MobiLink, set this parameter to 2.</p> <p>When set to 0, stored procedures are created for every call to SQLPrepare. This setting can result in decreased performance when processing statements that do not contain parameters.</p> <p>When set to 1 (the initial default), the driver creates stored procedures only if the statement contains parameters. Otherwise, the statement is cached and run directly at the time of SQLExecute.</p> <p>When set to 2, stored procedures are never created. The driver caches the statement, executes it directly at the time of SQLExecute, and reports any syntax or similar errors at the time of SQLExecute.</p> <p>When set to 3, stored procedures are never created. This is identical to value 2 except that any syntax or similar errors are returned at the time of SQLPrepare instead of SQLExecute. Use this setting only if you must have syntax errors reported at the time of SQLPrepare.</p>

Attribute	Description
PacketSize (PS)	<p>When set to -1, the driver computes the maximum allowable packet size on the first connect to the data source and saves the value in the system information.</p> <p>When set to 0 (the initial default), the driver uses the default packet size as specified in the Sybase server configuration.</p> <p>When set to x, an integer from 1 to 1024, the driver uses a packet size represented by x times 512 bytes. For example, PacketSize=6 means to set the packet size to 6 * 512 bytes (3072 bytes).</p> <p>To take advantage of this connection attribute, you must configure the Sybase server for a maximum network packet size greater than or equal to the value you specified for PacketSize. For example:</p> <pre data-bbox="705 807 1075 902">sp_configure "maximum network packet size", 5120 reconfigure Restart Sybase Server</pre> <p>NOTE: The ODBC specification specifies a connect option, SQL_PACKET_SIZE, that offers this same functionality. To avoid conflicts with applications that may set both the connection string attribute and the ODBC connect option, they have been defined as mutually exclusive. If PacketSize is specified, you will receive a message "Driver Not Capable" if you attempt to call SQL_PACKET_SIZE. If you do not set PacketSize, then application calls to SQL_PACKET_SIZE are accepted by the driver.</p>
Password (PWD)	A case-sensitive password.

Attribute	Description
RaiseErrorPositionBehavior (REPB)	<p>RaiseErrorPositionBehavior={0 1}. Specifies when the error is returned and where the cursor is positioned when raiserror is encountered.</p> <p>When set to 0 (the initial default), raiserror is handled separately from surrounding statements. The error is returned when raiserror is processed via SQLExecute, SQLExecDirect, or SQLMoreResults. The result set is empty.</p> <p>When set to 1 (MS compatible), raiserror is handled with the next statement. The error is returned when the next statement is processed and the cursor is positioned on the first row of subsequent result set. This could result in multiple raiserrors being returned on a single execute.</p>
SelectMethod (SM)	<p>SelectMethod={0 1}. Determines whether database cursors are used for SELECT statements.</p> <p>When set to 0 (the initial default), database cursors are used. In some cases performance degradation can occur when performing large numbers of sequential SELECT statements because of the amount of overhead associated with creating database cursors.</p> <p>When set to 1, SELECT statements are run directly without using database cursors, and the data source is limited to one active statement.</p> <p>To set the connection cache, SelectMethod must be set to 1. For more information, see CursorCacheSize, above.</p>

Attribute	Description
TightlyCoupled Distributed-Transactions (TCDT)	<p>TightlyCoupledDistributedTransactions={0 1}. Determines whether the driver uses tightly coupled distributed transactions when connected to an Adaptive Server Enterprise version 12 or higher database. When set to 1 (the initial default), the driver uses this type of transaction and multiple connections within the same distributed transaction do not obey each other's locks.</p> <p>When set to 0, the overall performance of the driver is better, but multiple connections within the same distributed transaction may hang each other because the connections do not obey each other's locks.</p> <p>This attribute is valid only when the driver is enlisted in a distributed transaction or when it is connected to a Sybase version 12 or higher database. Otherwise, this attribute is ignored.</p>
WorkstationID (WKID)	The workstation ID used by the client.
XAOpenStringParameters (XAOSP)	<p>Specifies trace file names. Use the syntax -Ltrace_filename, where trace_filename specifies the name of two trace files that will be created. The first trace file will trace all XA call activities and will be named exactly as you specified. The second trace file will contain tracing of any enlistment and unenlistment procedures and will be named as you specified with a "driver" extension. For example, if you specify XAtrace as the file name, the driver will create two trace files-XAtrace and XAtrace.driver.</p>

Data types

The following table shows how the Sybase data types are mapped to the standard ODBC data types.

Adaptive Server Enterprise	ODBC
binary	SQL_BINARY
bit	SQL_BIT
char	SQL_CHAR
datetime	SQL_TYPE_TIMESTAMP
decimal	SQL_DECIMAL
float	SQL_FLOAT
image	SQL_LONGVARBINARY
int	SQL_INTEGER
money	SQL_DECIMAL
numeric	SQL_NUMERIC
real	SQL_REAL
smalldatetime	SQL_TYPE_TIMESTAMP
smallint	SQL_SMALLINT
smallmoney	SQL_DECIMAL
sysname	SQL_VARCHAR
text	SQL_LONGVARCHAR
timestamp	SQL_TYPE_TIMESTAMP
tinyint	SQL_TINYINT
varbinary	SQL_VARBINARY
varchar	SQL_VARCHAR

Adaptive Server Enterprise 12.5

The *iAnywhere Solutions ODBC driver* supports extended new limits (XNL) for character and binary columns—columns with lengths greater than 255.

Unicode support

The Sybase Wire Protocol driver supports Unicode if the Sybase data source you are using to connect to the database is configured to use the UTF-8 character set. The data source is configured to use the UTF-8 character set by either setting the Charset attribute to UTF-8 in the connection string or in the `odbc.ini` file. See Table 14-1 for more information about the Charset attribute.

If the UTF-8 character set is installed, the driver maps the Sybase data types as follows:

ASE data type...	Mapped to...
Char	SQL_WCHAR
Unichar	SQL_WCHAR
Univarchar	SQL_WVARCHAR
Varchar	SQL_WVARCHAR
Text	SQL_WLONGVARCHAR

This driver supports the Unicode ODBC W (Wide) function calls, such as `SQLConnectW`. This allows the Driver Manager to transmit these calls directly to the driver. Otherwise, the Driver Manager would incur the additional overhead of converting the W calls to ANSI function calls, and vice versa.

Default Unicode mapping

The default Unicode mapping for an application's SQL_C_WCHAR variable is:

Platform	Default Unicode Mapping
Windows	UTF-16
AIX	UTF-8
HP-UX	UTF-8
Solaris	UTF-8
Linux	UTF-8

Supported features

This section lists features supported by the ODBC driver for Sybase Adaptive Server Enterprise.

Support for query timeout

The Adaptive Server Enterprise driver supports the QUERY_TIMEOUT statement attribute on Windows only.

Number of connections and statements supported

The Sybase database system supports multiple connections and multiple statements per connection. If SelectMethod=1, Sybase data sources are limited to one active statement in manual commit mode.

CHAPTER 3

iAnywhere Solutions ODBC Driver for Oracle

About this chapter

This chapter describes how to configure and use the iAnywhere Solutions ODBC driver for Oracle databases.

Contents

Topic:	page
Introduction	32
Key settings for MobiLink and Oracle	33
Configuring data sources	34
Connecting to a Data Source Using a Logon Dialog Box	37
Connecting to a data source using a connection string	38
Data types	45
Unicode support	46
Default Unicode Mapping	47
Stored procedure results	48
Supported features	49

Introduction

The iAnywhere Solutions ODBC driver for Oracle supports Oracle 8.1.6 and higher database systems (including Oracle 9i) on UNIX and Windows.

Driver requirements

The Oracle driver is a wire protocol driver, which means that there is no additional client software required.

Key settings for MobiLink and Oracle

There is one iAnywhere Solutions Oracle driver available: the Oracle Wire Protocol driver.

The iAnywhere Solutions Oracle Wire Protocol ODBC driver does not require Oracle client software.

The following are key settings for the Oracle wire protocol driver:

- ◆ **Basic connection settings** Set the following basic connection settings:
 - **Data source name** This value is used to identify the data source. On Windows, it is located on the General tab. On UNIX, this attribute is called DataSourceName (DSN).
 - **Host** This value provides the location of your Oracle server. On Windows, it is located on the General tab. On UNIX, this attribute is called HostName (HOST).
 - **Port** Identifies the port number of your Oracle listener. On Windows, it is located on the General tab. On UNIX, this attribute is called PortNumber (PORT).
 - **SID** Identifies the Oracle database system ID. On Windows, it is located on the General tab. On UNIX, this attribute is called Sid (SID).
 - **Default user name** This must correspond to a valid user name on the server. On Windows, it is located on the Advanced tab. On UNIX, this attribute is called LoginID (UID).
 - **Default Buffer Size for Long/LOB Columns(DLDBL)** This parameter specifies (in Kb) the size of the largest LONG BINARY or LONG VARCHAR column value being synchronized. The default setting is 1024, which is equivalent to 1 Mb. If you are synchronizing long columns with entries larger than this, set the value higher. This parameter appears on the Advanced tab of the Windows driver ODBC setup dialog. On UNIX, this attribute is called DefaultLongDataBuffLen (DLDBL).

☞ For complete information about settings for the Oracle driver, see [“Connecting to a data source using a connection string” on page 38.](#)

Configuring data sources

On Windows, data sources are configured and modified through the ODBC Administrator.

In the UNIX environment, there is no ODBC Administrator. To configure a data source in the UNIX environment, you must edit the system information file using the attributes in [“Values for IANAAppCodePage Connection String Attribute” on page 77](#). You must also edit this file to perform a translation.

☞ For more information, see [“The system information file \(.odbc.ini\)” on page 4](#).

❖ To configure an Oracle data source

1. Start the ODBC Administrator to display a list of data sources.
2. If you are configuring an existing data source, select the data source name and click Configure to display iAnywhere Solutions 9 - Oracle Wire Protocol Driver Setup.

If you are configuring a new data source, click Add to display a list of installed drivers. Select the Oracle Wire Protocol driver of your choice and click Finish to display the iAnywhere Solutions 9 - Oracle Wire Protocol Driver Setup.

NOTE: The General tab displays only fields that are required for creating a data source. The fields on all other tabs are optional, unless noted otherwise.

3. On the General tab, provide the following information and then click Apply.
 - ◆ **Data Source Name** Type a string that identifies this Oracle Wire Protocol data source configuration in the system information. Examples include “Accounting” or “Oracle-Serv1.”
 - ◆ **Description** Type an optional long description of a data source name. For example, “My Accounting Database” or “Oracle on Server number 1.”
 - ◆ **Host** Type either the name or the IP address of the server to which you want to connect. For example, if your network supports named servers, you can specify a server name such as Oracleserver. Or, you can specify an IP address such as 199.226.224.34.
 - ◆ **Port number** Type the port number of your Oracle listener. Check with your database administrator for the correct number.

- ◆ **SID** Type the Oracle System Identifier that refers to the instance of Oracle running on the server.
- 4. Optionally, click the Advanced tab to specify data source settings. On this tab, provide any of the following optional information; then, click Apply.
 - ◆ **Default User Name** Type the default user name used to connect to your Oracle database. A default user name is required only if security is enabled on your database. Your ODBC application may override this value or you may override this value in the logon dialog box or connection string.
 - ◆ **Array Size** Type the number of bytes the driver uses for fetching multiple rows. Values can be an integer from 1 up to 4 GB; the default is 60000. Larger values increase throughput by reducing the number of times the driver fetches data across the network. Smaller values increase response time, as there is less of a delay waiting for the server to transmit data.
 - ◆ **Lock Timeout** Type 0, -1, or any integer value greater than 0. The value 0 specifies that Oracle does not wait for a lock to be freed before raising an error when processing a Select...For Update statement. The value -1 waits forever. When connected to an Oracle9i server, you can specify the number of seconds to wait by setting this option to an integer greater than 0. If you are connected to an Oracle8i or lower server, any value greater than 0 is equivalent to the value -1.
 - ◆ **Default Buffer Size for Long/LOB Columns (in Kb)** Type an integer value that specifies the maximum length of data fetched from Long/LOB columns. The value must be in multiples of 1024 (for example, 1024, 2048). The default is 1024 KB. You will need to increase this value if the total size of any long data exceeds 1 MB.
 - ◆ **Local Timezone Offset** This feature is not supported. Do not enter anything.
 - ◆ **Enable Timestamp with Timezone** This feature is not supported. Do not select it. By default, the check box is not selected.
 - ◆ **Catalog Options** Select this check box if you want the result column REMARKS for the catalog functions SQLTables and SQLColumns, and the result column COLUMN_DEF for the catalog function SQLColumns to have meaning for Oracle. Selecting this box reduces the performance of your queries. By default, the check box is not selected, which returns SQL_NULL_DATA for the result columns COLUMN_DEF and REMARKS.
 - ◆ **Enable SQLDescribeParam** Select this check box to enable the SQLDescribeParam function, which results in all parameters being described with a data type of SQL_VARCHAR. This option should be

selected when using Microsoft Remote Data Objects (RDO) to access data. By default, the check box is not selected.

- ◆ **Application Using Threads** This check box must be selected. Selecting this option ensures that the driver works with multi-threaded applications. By default, the check box is selected.
- ◆ **Procedure Returns Results** Select this check box to enable the driver to return result sets from stored procedures/functions. If this check box is selected and you execute a stored procedure that does not return result sets, you will incur a small performance penalty. By default, the check box is not selected.
- ◆ **Enable Static Cursors for Long Data:** Select this check box to enable the driver to support long columns when using a static cursor. Selecting this check box causes a performance penalty at the time of execution when reading long data. By default, the check box is not selected.
NOTE: You must select this check box if you want to persist a result set that contains LONG data into an XML data file.
- ◆ **Use Current Schema for SQLProcedures** Select this check box to specify only the current user when executing SQLProcedures. When this check box is selected (the default), the call for SQLProcedures is optimized, but only procedures owned by the user are returned.
- ◆ **Catalog Functions Include Synonyms** By default, the check box is selected.
- ◆ **Describe at Prepare** Select this check box to enable the driver to describe the SQL statement at prepare time. By default, the check box is not selected.
- ◆ **Translate** Click Translate to display the Select Translator dialog box, which lists the translators specified in the ODBC Translators section of the system information. DataDirect provides a translator named “OEM TO ANSI” that translates your data from the IBM PC character set to the ANSI character set. Select a translator; then, click OK to close this dialog box and perform the translation.

5. At any point during the configuration process, you can click Test Connect to attempt to connect to the data source using the connection properties specified in the Driver Setup dialog box. A logon dialog box is displayed. Note that the information you enter in the logon dialog box during a test connect is not saved. If the driver can connect, it releases the connection and displays a “connection established” message. Click OK. If the driver cannot connect because of an improper environment or incorrect connection value, it will display an appropriate error message. Click OK.

Connecting to a Data Source Using a Logon Dialog Box

Some ODBC applications display a logon dialog box when you are connecting to a data source. In these cases, the data source name has already been specified.

In this dialog box, perform the following steps:

In the Host field, type either the name or the IP address of the server to which you want to connect.

In the Port Number field, type the number of your Oracle listener. Check with your database administrator for the correct number.

In the SID field, type the Oracle System Identifier that refers to the instance of Oracle running on the server.

If required, type your Oracle user name.

If required, type your Oracle password.

Click OK to log on to the Oracle database installed on the server you specified and to update the values in the system information.

NOTE: Oracle has a feature that allows you to connect to Oracle via the operating system user name and password. To connect, use a slash (/) for the user name and leave the password blank. To configure the Oracle server, refer to the Oracle server documentation.

Connecting to a data source using a connection string

If your application requires a connection string to connect to a data source, you must specify the data source name that tells the driver which section in the system information to use for the default connection information. Optionally, you may specify `attribute=value` pairs in the connection string to override the default values stored in the system information. These values are not written to the system information.

You can specify either long or short names in the connection string. The connection string has the form:

```
DSN=data_source_name[:attribute=value[:attribute=value]. . . ]
```

An example of a connection string for Oracle is:

```
DSN=Accounting;SRVR=QESRVR;UID=JOHN;PWD=XYZZY
```

If the server name contains a semicolon, enclose it in quotation marks:

```
DSN=Accounting;SRVR="QE;SRVR";UID=JOHN;PWD=XYZZY
```

To configure a data source in the UNIX environment, you must edit the system information file. This file accepts only long names for attributes.

☞ For more information, see [“Quick start for UNIX” on page 4](#).

The defaults listed in the table are initial defaults that apply when no value is specified in either the connection string or in the data source definition in the system information. If you specified a value for the attribute when configuring the data source, that value is the default.

☞ Most settings can be left at their default values. For a list of key driver settings when using MobiLink, see [“Key settings for MobiLink and Oracle” on page 33](#).

Attribute	Description
AppCodePage (ACP)	AppCodePage has been replaced by IANAAppCodePage (IACP). See below.

Attribute	Description
ApplicationUsingThreads (AUT)	<p>ApplicationUsingThreads={0 1}. Ensures that the driver works with multi-threaded applications.</p> <p>When set to 1 (the initial default), the driver is thread-safe.</p> <p>When using the driver with single-threaded applications, you can set this option to 0 to avoid additional processing required for ODBC thread-safety standards.</p> <p>Set to 1 for DSNs used with MobiLink server.</p>
ArraySize (AS)	<p>The number of bytes the driver uses for fetching multiple rows. Values can be an integer from 1 up to 4 GB. Larger values increase throughput by reducing the number of times the driver fetches data across the network. Smaller values increase response time, as there is less of a delay waiting for the server to transmit data.</p> <p>The initial default is 60,000.</p>
CachedCursorLimit (CCL)	<p>The number of Oracle Cursor Identifiers that the driver stores in cache. A cursor identifier is needed for each open result set. When a result set is closed the driver stores the identifier in its cache, up to the limit specified, rather than tell the Oracle database server to close the cursor identifier. When a new cursor identifier is needed, the driver takes one from its cache, if one is available. Cached cursor identifiers are closed when the connection is closed. The initial default is 32.</p>

Attribute	Description
CachedDescriptionLimit (CDL)	<p>The number of descriptions, from 0 to 65535, that the driver saves for SELECT statements. These descriptions include number of columns, data type, length, and scale for each column. The matching is done by an exact-text match through the FROM clause. The default is 0 because of the overhead of parsing the SQL statement. Using the FROM clause, applications that issue a SELECT statement that returns a few rows repeatedly can realize a significant performance benefit when using this option.</p> <p>NOTE: If the statement contains a UNION or a nested SELECT, the description is not cached.</p>
CatalogIncludesSynonyms (CIS)	<p>CatalogIncludesSynonyms={0 1}. Determines whether synonyms are included in calls to SQLProcedures, SQLStatistics, and SQLProcedureColumns. When set to 1, synonyms are included in these calls. When set to 0, synonyms are not included (a non-standard behavior) and performance is improved.</p> <p>The ODBC specification requires that synonyms be included in the result sets of these catalog calls when the calls are unqualified. Most users do not qualify queries; therefore, synonyms are included by default, which slows down performance. This option allows you to improve performance by not including synonyms. The initial default is 1.</p>
CatalogOptions (CO)	<p>CatalogOptions={0 1}. Determines whether the result column REMARKS for the catalog functions SQLTables and SQLColumns and COLUMN_DEF for the catalog function SQLColumns have meaning for Oracle. If you want to obtain the actual default value, set CO=1. The initial default is 0.</p>
DataSourceName (DSN)	<p>A string that identifies an Oracle data source configuration in the system information. Examples include "Accounting" or "Oracle-Serv1."</p>

Attribute	Description
DefaultLongDataBufLen (DLDBL)	An integer value that specifies, in 1024-byte multiples, the maximum length of data fetched from Long/LOB columns. You will need to increase this value if the total size of any long data exceeds 1 MB. The default is 1024.
DescribeAtPrepare (DAP)	DescribeAtPrepare={0 1}. Determines whether the driver describes the SQL statement at prepare time. When set to 0 (the initial default), the driver does not describe the SQL statement at prepare time.
EnableDescribeParam (EDP)	EnableDescribeParam={0 1}. Determines whether the ODBC API function SQLDescribeParam is enabled, which results in all parameters being described with a data type of SQL_VARCHAR. This attribute should be set to 1 when using Microsoft Remote Data Objects (RDO) to access data. The initial default is 0.
EnableScrollableCursors (ESC)	EnableScrollableCursors={0 1}. Enables scrollable cursors for the data source. Both Keyset and Static cursors are enabled. The initial default is 1.
EnableStaticCursorsForLongData (ESCLD)	EnableStaticCursorsForLongData={0 1}. Determines whether the driver supports long columns when using a static cursor. Using this attribute causes a performance penalty at the time of execution when reading long data. The initial default is 0.
EnableTimestampWithTimezone (ETWT)	EnableTimestampWithTimezone={0 1}. Determines whether the driver exposes timestamps with time zones to the application. When set to 1, the driver issues an “ALTER SESSION” at connection time to modify NLS_TIMESTAMP_TZ_FORMAT. NLS_TIMESTAMP_TZ_FORMAT is changed to the ODBC definition of a timestamp literal with the addition of the timezone literal: ‘YYYY-MM-DD HH24:MI:SSXFF TZR’. The initial default is 0.

Attribute	Description
HostName (HOST)	<p>HostName={servername IP_address}. Identifies the Oracle server to which you want to connect. If your network supports named servers, you can specify a host name such as Oracle-server. Otherwise, specify an IP address such as 199.226.224.34.</p>
IANAAppCodePage (IACP)	<p>Valid values for this attribute are listed in “Values for IANAAppCodePage Connection String Attribute” on page 77. The code page that you specify must be the same as the code page used by your application. The driver on UNIX determines the value of the application’s code page by checking for an IANAAppCodePage value in the following order:</p> <ul style="list-style-type: none"> ◆ In the connection string ◆ In the DataSource section of the system file (odbc.ini) ◆ In the ODBC section of the system file (odbc.ini) <p>If no IANAAppCodePage value is found, the driver uses the default value of 4 (ISO 8859-1 Latin-1).</p> <p>NOTE: The IANAAppCodePage connection string attribute replaces the AppCodePage connection string attribute in earlier versions of Connect for ODBC. The drivers are backwardly compatible with the AppCodePage</p>
LocalTimeZoneOffset (LTZO)	<p>The default for this attribute is “” (empty string). This means that the driver determines local time zone information from the operating system. If it is not available from the operating system, the driver defaults to using the setting on the Oracle server. Other possible values are specified as offsets from GMT in the following format: (-)HH:MM. For example, “-08:00” equals GMT minus 8 hours. The driver uses the value of the attribute to issue an “ALTER SESSION” for the local time zone at connect time.</p>

Attribute	Description
LockTimeOut (LTO)	<p>LockTimeOut={0 -1}. Determines whether Oracle should wait for a lock to be freed before raising an error when processing a Select... For Update statement.</p> <p>When set to 0, Oracle does not wait.</p> <p>When set to -1 (the initial default), Oracle waits forever.</p>
LogonID (UID)	<p>The default logon ID (user name) that the application uses to connect to your Oracle database. A logon ID is required only if security is enabled on your database. If so, contact your system administrator to get your logon ID.</p>
Password (PWD)	<p>The password that the application uses to connect to your Oracle database.</p>
PortNumber (PORT)	<p>Identifies the port number of your Oracle listener. The initial default value is 1521. Check with your database administrator for the correct number.</p>
ProcedureRetResults (PRR)	<p>ProcedureRetResults={0 1}. Determines whether the driver returns result sets from stored procedure functions.</p> <p>When set to 0 (the initial default), the driver does not return result sets from stored procedures.</p> <p>When set to 1, the driver returns result sets from stored procedures. When set to 1 and you execute a stored procedure that does not return result sets, you will incur a small performance penalty. For details, see “Stored procedure results” on page 48.</p>
Sid (SID)	<p>The Oracle System Identifier that refers to the instance of Oracle running on the server. This item is required when connecting to servers that support more than one instance of an Oracle database.</p>

Attribute	Description
UseCurrentSchema (UCS)	<p data-bbox="673 274 1154 361">UseCurrentSchema={0 1}. Determines whether the driver specifies only the current user when executing SQLProcedures.</p> <p data-bbox="673 369 1154 430">When set to 0, the driver does not specify only the current user.</p> <p data-bbox="673 439 1154 534">When set to 1 (the initial default), the call for SQLProcedures is optimized, but only procedures owned by the user are returned.</p>

Data types

The following table shows how the Oracle data types are mapped to the standard ODBC data types.

Oracle	ODBC
Bfile	SQL_LONGVARBINARY ^{1,2}
BLOB	SQL_LONGVARBINARY ²
Char	SQL_CHAR
CLOB	SQL_LONGVARCHAR ²
Date	SQL_TYPE_TIMESTAMP
Long	SQL_LONGVARCHAR
Long Raw	SQL_LONGVARBINARY
Number	SQL_DOUBLE
Number(p,s)	SQL_DECIMAL
Raw	SQL_VARBINARY
Varchar ¹	SQL_VARCHAR

¹ Valid when connecting to Oracle 8 servers; these data types support output parameters to stored procedures.

The Oracle Wire Protocol driver does not support any Abstract Data Types. When the driver encounters an Abstract Data Type during data retrieval, it will return an Unknown Data Type error (SQL State HY000).

Unicode support

The Oracle Wire Protocol driver automatically determines whether the Oracle database is a Unicode database.

The Oracle Wire Protocol driver maps the Oracle data types as follows:

Oracle Data Type	Mapped to
Char	SQL_WCHAR
Varchar2	SQL_WVARCHAR
Long	SQL_WLONGVARCHAR
Clob	SQL_WLONGVARCHAR

This driver supports the Unicode ODBC function calls, called W (Wide) calls (for example, SQLConnectW). These calls are used to accept Unicode datastreams.

Default Unicode Mapping

The default Unicode mapping for an application's SQL_C_WCHAR variable is as follows:

Platform	Default Unicode Mapping
Windows	UTF-16
AIX	UTF-8
Solaris	UTF-8
Linux	UTF-8

Stored procedure results

When the connection option `ProcedureRetResults` is active, the driver returns result sets from stored procedures/functions. In addition, `SQLGetInfo(SQL_MULT_RESULTS_SETS)` will return `Y` and `SQLGetInfo(SQL_BATCH_SUPPORT)` will return `SQL_BS_SELECT_PROC`. If this option is on and you execute a stored procedure that does not return result sets, you will incur a small performance penalty.

This feature requires that stored procedures be in a certain format. First, a package must be created to define all of the cursors used in the procedure, then the procedure can be created using the new cursor. For example:

```
Create or replace package GEN_PACKAGE as
  CURSOR G1 is select CHARCOL from GTABLE2;
  type GTABLE2CHARCOL is ref cursor return G1%rowtype;
end GEN_PACKAGE;
Create or replace procedure GEN_PROCEDURE1 (
  rset IN OUT GEN_PACKAGE.GTABLE2
  CHARCOL, icol INTEGER) as
begin
  open rset for select CHARCOL from GTABLE2
    where INTEGERCOL <= icol order by INTEGERCOL;
end;
```

When executing the stored procedures with result sets, do not include the result set arguments in the list of procedure arguments. The previously described example would be executed as:

```
{call GEN_PROCEDURE1 (?)}
```

where `?` is the parameter for the `icol` argument.

☞ For more information, consult your Oracle SQL manual.

Supported features

This section lists features supported by the iAnywhere Solutions 9 — Oracle Wire Protocol ODBC driver.

Isolation and lock levels supported

Oracle supports isolation level 1 (read committed) and isolation level 3 (serializable isolation). Oracle supports record-level locking.

ODBC conformance level

☞ For a list of the API functions supported by the Oracle Wire Protocol driver, see [“ODBC Functions” on page 69](#).

The Oracle Wire Protocol driver also support the following functions:

- ◆ SQLColumnPrivileges
- ◆ SQLDescribeParam (if EnableDescribeParam=1)
- ◆ SQLForeignKeys
- ◆ SQLPrimaryKeys
- ◆ SQLProcedures
- ◆ SQLProcedureColumns
- ◆ SQLSetPos
- ◆ SQLTablePrivileges

The drivers support the core SQL grammar.

Number of connections and statements supported

The Oracle driver supports multiple connections and multiple statements per connection.

CHAPTER 4

iAnywhere Solutions ODBC Driver for DB2

About this chapter

This chapter describes how to configure and use the iAnywhere Solutions ODBC driver for IBM DB2 databases.

Contents

Topic:	page
Introduction	52
Key settings for MobiLink and DB2	53
Configuring data sources	54
Connecting to a data source using a logon dialog box	59
Connecting to a data source using a connection string	60
Data types	65
Supported features	66
Unicode support	67
Default Unicode mapping	68

Introduction

The iAnywhere Solutions ODBC driver for DB2 (the “DB2 driver”) supports DB2 Universal Database versions 7 and 8 on UNIX and Windows.

Note: When used with DB2 UDB versions 7.1 or 7.2, this driver does not support the DB2 BLOB data type, and only supports 32 Kb CLOB data. If you are using these data types with DB2 7.1 or 7.2, you should use the IBM DB2 UDB 7.2 driver (version 7.01.00.65 or 7.02.00.05).

Driver requirements

The server requirement for all platforms is the same. The UDB DB2 database must be installed as the Server Version (not the Local Version).

The DB2 driver is a wire protocol driver, which means that there is no additional client software required.

Key settings for MobiLink and DB2

The iAnywhere Solutions ODBC driver for IBM DB2 does not require any DB2 client software. The following are key settings for DB2. Other parameters are available and described elsewhere, but are of less importance.

- ◆ **Basic connection settings** Set the following basic connection settings:
 - **Data Source Name (DSN)** This value is used to identify the data source.
 - **Ip Address (IP)** The IP address of the machine where the catalog tables are stored.
 - **Default User ID (UID)** This default user ID must correspond to a valid user name on the server.
 - **Tcp Port** The port number of the DB2 DRDA listener process on the server host.
 - **Database Name** The name of the database to which you want to connect.
- ◆ **WithHold (WH)** For use with MobiLink, this attribute must be set to 1 in the ODBC configuration file.

☞ For complete information about settings for DB2, see [“Connecting to a data source using a connection string” on page 60](#).

Bind packages

Users must create the DB2 bind packages on every server to which they intend to connect with the driver. The driver will not work properly with any server that does not have the packages created. The UNIX version of the driver is provided with a program that creates the bind package. It is the equivalent of the Create Package button on the Bind tab of the DB2 driver setup on Windows.

The bind program is shipped with SQL Anywhere Studio, and can be found in the *drivers/lib* directory under your SQL Anywhere directory. You must export `LD_LIBRARY_PATH` or `LIBPATH` before creating bind packages.

☞ For more information, see [“Configuring data sources” on page 54](#).

Configuring data sources

To configure a data source in the UNIX environment, you must edit the system information file using the attributes in [“Values for IANAAppCodePage Connection String Attribute” on page 77](#). You must also edit this file to perform a translation.

☞ For more information, see [“Quick start for UNIX” on page 4](#).

Creating DB2 bind packages

Users must create the DB2 bind packages on every server to which they intend to connect with the driver. The driver will not work properly with any server that does not have the packages created. The UNIX version of the driver is provided with a program that creates the bind package. It is the equivalent of the Create Package button on the Bind tab of the DB2 driver setup.

The bind program is shipped with SQL Anywhere Studio, and can be found in the *drivers/lib* directory under your SQL Anywhere directory. You must export LD_LIBRARY_PATH or LIBPATH before creating bind packages.

❖ To bind a package (UNIX)

1. At a command prompt, enter the following command:

```
bind19 dsn
```

where *dsn* is the ODBC data source name. You are prompted for a user ID and password if they are not stored in the system information file.

❖ To configure a DB2 Wire Protocol data source (Windows)

1. Start the ODBC Administrator to display a list of data sources.
2. If you are configuring an existing data source, select the data source name and click Configure to display the ODBC DB2 Wire Protocol Driver Setup dialog box.

If you are configuring a new data source, click Add to display a list of installed drivers. Select the DB2 Wire Protocol driver and click Finish to display the ODBC DB2 Wire Protocol Driver Setup dialog box.

NOTE: The General and Bind tabs display only fields that are required for creating a data source. The fields on other tabs are optional, unless noted otherwise.

3. On the General tab, provide the following information and click Apply.
 - ◆ **Data Source Name** Type a string that identifies this DB2 data source configuration in the system information. If you are creating a new data source definition, type a unique name of up to 32 characters. If you

specify the name of an existing data source definition, the new settings will replace the existing ones.

- ◆ **Description** Type an optional descriptive comment for this data source definition. ODBC-related applications and development tools often display this description with the data source name when they display a list of data sources. If you want to include a description for this data source definition, type a comment of up to 64 characters.
 - ◆ **Ip Address** Type the IP (Internet Protocol) address of the machine where the catalog tables are stored. Specify the address using the machine's numeric address (for example, 123.456.78.90) or specify its host name. If you enter a host name, the driver must find this name (with the correct address assignment) in the HOSTS file on the workstation or in a DNS server.
 - ◆ **Tcp Port** Type the port number that is assigned to the DB2 DRDA listener process on the server host machine. Specify either this port's numeric address or its service name (5179 is the default port address). If you specify a service name, the driver must find this name (with the correct port assignment) in the SERVICES file on the workstation.
 - ◆ **Location Name** This field is valid only if you are connecting to a DB2 database running on OS/390 or AS/400. This field is disabled if the Database Name field is populated.
 - ◆ **Collection** This field is valid only if you are connecting to a DB2 database running on OS/390 or AS/400. This field is disabled if the Database Name field is populated.
 - ◆ **Database Name** This field is valid only if you are connecting to a DB2 database running on UNIX or NT. Type the name of the database to which you want to connect. NOTE: This field is disabled if the Location Name field is populated.
 - ◆ **Package** Type the name of the package that the driver uses to process dynamic SQL for applications that use this data source definition. The default name is DDODBC.
 - ◆ **Default User ID** Type the default logon ID used to connect to your DB2 database. Your ODBC application may override this value or you may override it in the logon dialog box or connection string. This field is optional.
4. Optionally, click the Advanced tab to specify data source settings. On this tab, provide any of the following optional information and click Apply.
- ◆ **With Hold Cursors** This option specifies the cursor behavior for the application used with this data source-either DB2 closes all open

cursors (Delete cursors) after a commit or rollback or leaves them open (Preserve cursors). When this check box is selected (the default), the cursor behavior is Preserve (SQLGetInfo() returns SQL_CB_PRESERVE for SQL_COMMIT_CURSOR_BEHAVIOR). When the check box is not selected, the cursor behavior is Delete (SQLGetInfo() returns SQL_CB_DELETE). For information about this function, refer to the Microsoft ODBC API.

- ◆ **Add to Create Table** Type a string that is automatically added to all Create Table statements. This field is primarily for users who need to add an “in database” clause.
- ◆ **Alternate ID** Type a value to be substituted at connect time for the current schema. This sets the default qualifier for unqualified object names in SQL statements. If the attempt to change the current schema fails, the connection fails with “Invalid value for Alternate ID.” DB2 permissions should be set to SYSADM. (Not valid for AS/400 V4R5 and V5R1.)
- ◆ **Translate** Click Translate to display the Select Translator dialog box, which lists the translators specified in the ODBC Translators section of the system information. DataDirect provides a translator named “OEM to ANSI” that translates your data from the IBM PC character set to the ANSI character set.
Select a translator and click OK to close this dialog box and perform the translation.

5. Click the Bind tab to configure options for creating bind packages.

The Bind tab allows you to create the bind packages on the server that will be used by the driver. The tab also allows you to specify the behavior of the package. You must create the bind packages on every server to which you intend to connect with the driver. The driver will not work properly with any server that does not have the packages created.

6. On the Bind tab, provide the following information and click Apply.

- ◆ **Grant Execute** Select this check box to grant execute privileges on the package that you are creating. The default value is grant execute privileges on the package to PUBLIC. You can also specify to whom to grant execute privileges.
- ◆ **Isolation Level** Select the method by which locks are acquired and released by the system. Valid values are:
 - **All** Prevents any other process from accessing data that your application has read or modified. All read or modified data is locked until the end of the transaction.

- **Change** Allows other processes to read from the database. Only modified data is locked until the end of the transaction.
 - **Cursor Stability (the default)** Allows other processes to change a row that your application has read if the cursor is not on the row you want to change. Prevents other processes from changing records that your application has changed until your program commits them or terminates. Prevents your program from reading a modified record that has not been committed by another process.
 - **No Commit** Allows your program to read modified records even if they have not been committed by another person. (On AS/400, this is the only the only isolation level that works for collections that do not have journaling enabled.)
 - **Repeatable Read** Prevents other processes from changing records that are read or changed by your application (including phantom records) until your program commits them or terminates. Prevents the application from reading modified records that have not been committed by another process. If your program opens the same query during a single unit of work under this isolation level, the results table will be identical to the previous table; however, it can contain updates made by your program.
 - ◆ **Package Owner** Package Owner: Type the AuthID assigned to the package. This DB2 AuthID must have authority to execute all the SQL in the package.
 - ◆ **Dynamic Sections** Type the number of statements that the DB2 Wire Protocol driver package can prepare for a single user. The default is 100.
 - ◆ **Create Package** Click to configure a package. When you click the Create Package button, a logon dialog is displayed. Enter your user ID and password; then, click Login. A message similar to the following is displayed if the package was not created successfully:

```
Program name DDODBC not found in plan. Please bind the  
required packages through the DB2 driver setup screen
```
7. At any point during the configuration process, you can click Test Connect to attempt to connect to the data source using the connection properties specified in the Driver Setup dialog box. A logon dialog box is displayed; see [“Connecting to a data source using a logon dialog box” on page 59](#). Note that the information you enter in the logon dialog box during a test connect is not saved.
- ◆ If the driver can connect, it releases the connection and displays a “connection established” message. Click OK.

-
- ◆ If the driver cannot connect because of an improper environment or incorrect connection value, it will display an appropriate error message. Click OK.
8. Click OK or Cancel. If you click OK, the values you have specified become the defaults when you connect to the data source. You can change these defaults by using this procedure to reconfigure your data source. You can override these defaults by connecting to the data source using a connection string with alternate values.

Connecting to a data source using a logon dialog box

On Windows, some ODBC applications display a logon dialog box when you are connecting to a data source. In these cases, the data source name has already been specified. For DB2, the dialog box is as follows:

In this dialog box, perform the following steps:

1. In the IP Address field, type the IP address of the machine where the catalog tables are stored. Specify the address using the machine's numeric address (for example, 123.456.78.90) or specify its host name. If you enter a host name, the driver must find this name (with the correct address assignment) in the HOSTS file on the workstation or in a DNS server.
2. In the TCP Port field, type the port number that is assigned to the DB2 server on the machine where the catalog tables are stored. Specify either this port's numeric address or its service name (5179 is the default port address). If you specify a service name, the driver must find this name (with the correct port assignment) in the SERVICES file on the workstation.
3. Click OK to complete the logon and to update the values in the system information.

Connecting to a data source using a connection string

If your application requires a connection string to connect to a data source, you must specify the data source name that tells the driver which section in the system information to use for the default connection information. Optionally, you may specify attribute=value pairs in the connection string to override the default values stored in the system information. These values are not written to the system information.

You can specify either long or short names in the connection string. The connection string has the form:

DSN=data_source_name[;attribute=value[;attribute=value]. . .]

An example of a connection string for DB2 is:

```
DSN=DB2MVS ; LOC=TESTMVSDDB2 ; UID=JOHN ; PWD=XYZZY
```

To configure a data source in the UNIX environment, you must edit the system information file. This file accepts only long names for attributes.

☞ For more information, see [“Quick start for UNIX” on page 4](#).

The defaults listed in the table are initial defaults that apply when no value is specified in either the connection string or in the data source definition in the system information. If you specified a value for the attribute when configuring the data source, that value is the default.

☞ Most settings can be left at their default values. For a list of key driver settings when using MobiLink, see [“Key settings for MobiLink and DB2” on page 53](#). For other information, about using ODBC drivers with MobiLink, see http://www.ianywhere.com/developer/technotes/odbc_mobilink.html.

Attribute	Description
AddStringToCreateTable (ASCT)	A string that is automatically added to all Create Table statements. This field is primarily for users who need to add an “in database” clause.
AlternateID (AID)	This connection string changes the current schema to a user-specified value at connect time. It sets the default qualifier for unqualified object names in SQL statements. If the attempt to change the current schema fails, the connection fails with “Invalid value for Alternate ID.” DB2 permissions should be set to SYSADM. (Not valid for AS/400 V4R5 and V5R1.)

Attribute	Description
AppCodePage (ACP)	AppCodePage has been replaced by IANAAppCodePage (IACP). See below.
Collection (COL)	A name that identifies a group of packages. This attribute is valid only if you are connecting to DB2 on OS/390.
Database (DB)	The name of the database to which you want to connect.
DataSource-Name (DSN)	A string that identifies a DB2 data source configuration in the system information. Examples include “Accounting” and “DB2-Serv1”.
DynamicSections (DS)	The number of statements that the DB2 driver package can prepare for a single user. The initial default is 32.
GrantAuthid (GA)	<p>A value that determines to whom execute privileges are granted.</p> <p>The default value is grant execute privileges on the package to PUBLIC.</p>
GrantExecute (GE)	<p>GrantExecute={0 1}. Indicates whether or not to grant privileges on the package that you are creating.</p> <p>When set to 0, privileges are not granted.</p> <p>When set to 1, privileges are granted.</p>
IANAAppCodePage (IACP)	<p>Valid values for this attribute are listed in “Values for IANAAppCodePage Connection String Attribute” on page 77. The code page that you specify must be the same as the code page used by your application. The driver on UNIX determines the value of the application’s code page by checking for an IANAAppCodePage value in the following order:</p> <ul style="list-style-type: none"> ◆ In the connection string ◆ In the DataSource section of the system file (odbc.ini) ◆ In the ODBC section of the system file (odbc.ini) <p>If no IANAAppCodePage value is found, the driver uses the default value of 4 (ISO 8859-1 Latin-1).</p> <p>NOTE: The IANAAppCodePage connection string attribute replaces the AppCodePage connection string attribute in earlier versions of Connect for ODBC. The drivers are backwardly compatible with the AppCodePage attribute, but you must now use the IANAAppCodePage attribute.</p>

Attribute	Description
IPAddress (IP)	The IP address of the machine where the catalog tables are stored. Enter the address using the machine's numeric address (for example, 123.456.78.90) or type its address name. If you enter an address name, the driver must find this name (with the correct address assignment) in the HOSTS file on the workstation or in a DNS server.
IsolationLevel (IL)	<p>IsolationLevel={ ALL CHANGE CURSOR_-STABILITY NO_COMMIT REPEATABLE_READ}.</p> <p>Specifies the method by which locks are acquired and released by the system. Valid values are:</p> <p>All Prevents any other process from accessing data that your application has read or modified. All read or modified data is locked until the end of the transaction.</p> <p>Change Allows other processes to read from the database. Only modified data is locked until the end of the transaction.</p> <p>Cursor Stability Allows other processes to change a row that your application read if the cursor is not on the row that you want to change. Prevents other processes from changing records that your application has changed until your program commits them or terminates. Prevents your program from reading a modified record that has not been committed by another process.</p>
IsolationLevel (IL) (cont.)	<p>No Commit Allows your program to read modified records even if they have not been committed by another person.</p> <p>Repeatable Read Prevents other processes from changing records that are read or changed by your application (including phantom records) until your program commits them or terminates. Prevents the application from reading modified records that have not been committed by another process. If your program opens the same query during a single unit of work under this isolation level, the results table will be identical to the previous table; however, it can contain updates made by your program.</p>
Location (LOC)	A path that specifies the DB2 location name. This attribute is valid and required only if you are connecting to DB2 on OS/390.

Attribute	Description
LogonID (UID)	<p>The default logon ID used to connect to your DB2 database. A logon ID is required only if security is enabled on your database. If so, contact your system administrator to get your logon ID.</p> <p>For DB2 on UNIX, normal UNIX security is used. The LogonID value is your UNIX user ID.</p>
Package (PCK)	<p>The name of the package that the driver uses to process static and dynamic SQL for applications that use this data source definition. The default name is DEFxx, where xx is the version number.</p>
PackageOwner (PO)	<p>The AuthID assigned to the package. This DB2 AuthID must have authority to execute all the SQL in the package.</p>
Password (PWD)	<p>A password used to connect to your DB2 database.</p>
TcpPort (PORT)	<p>The port number that is assigned to the DB2 server on the machine where the catalog tables are stored. Specify this port's numeric address or its name (5179 is the default port address). If you specify a port name, the driver must find this name (with the correct port assignment) in the SERVICES file on the workstation.</p>

Attribute	Description
WithHold (WH)	<p data-bbox="584 276 1013 296"><i>Set this attribute to 1 for use with MobiLink.</i></p> <p data-bbox="584 314 1143 482">WithHold={0 1}. Specifies the cursor behavior for the application used with this data source. Either DB2 closes all open cursors (Delete cursors) after a commit or rollback, or leaves them open (Preserve cursors). When set to 1, the cursor behavior is Preserve. When set to 0, the cursor behavior is Delete (the default).</p> <p data-bbox="584 499 1143 609">If you are using the Static Bind Administrator and you want your package to use cursors WITH HOLD, you must set this attribute to 1. Note that any application using this package must use a data source with this attribute set to 1.</p> <p data-bbox="584 626 1143 795">When set to 1, the Static Bind Administrator automatically adds the WITH HOLD clause to queries that it puts in the application's database resource module (DBRM). The WITH HOLD clause prevents DB2 from automatically closing the cursor when the application executes a Commit statement.</p> <p data-bbox="584 812 1018 878"><code>SQLGetInfo()</code> returns <code>SQL_CB_PRESERVE</code> for <code>SQL_COMMIT_CURSOR_BEHAVIOR</code></p> <p data-bbox="584 895 1143 987">When set to 0, <code>SQLGetInfo()</code> returns <code>SQL_CB_DELETE</code>. For information about this function, refer to the Microsoft ODBC API.</p>

Data types

The following table shows how the DB2 data types map to the standard ODBC data types.

DB2	ODBC
Bigint ¹	SQL_BIGINT
Blob ²	SQL_LONGVARBINARY
Char	SQL_CHAR
Char() for Bit Data	SQL_BINARY
CLOB ³	SQL_LONGVARCHAR
Date	SQL_TYPE_DATE
Decimal	SQL_DECIMAL
Float	SQL_DOUBLE
Integer	SQL_INTEGER
Long Varchar	SQL_LONGVARCHAR
Long Varchar for Bit Data	SQL_LONGVARBINARY
Smallint	SQL_SMALLINT
Time	SQL_TYPE_TIME
Timestamp	SQL_TYPE_TIMESTAMP
Varchar	SQL_VARCHAR
Varchar() for Bit Data	SQL_VARBINARY

1 - Supported on DB2 Universal Database 8.1.

2 - Supported on DB2 Universal Database 8.1.

3 - On DB2 Universal Database versions previous to 8.1, only the first 32Kb of the Clob data type are returned when fetching, and only 32Kb can be inserted and updated.

Supported features

This section lists features supported by the ODBC driver for DB2.

Stored procedure support

The DB2 driver supports DB2 Remote Procedure Calls (RPCs) with the following restrictions:

- ◆ Multiple result sets are returned.
- ◆ RPCs must take an argument list. The driver does not support RPCs that use a SQL descriptor area (SQLDA) data structure to specify the arguments.
- ◆ Literals are supported as stored procedure parameters.

Isolation and lock levels supported

DB2 supports isolation levels 0 (read uncommitted), 1 (read committed), and 2 (repeatable read). It supports record-level locking.

NOTE: An isolation level can be set only before connecting to a DB2 database.

Number of connections and statements supported

The DB2 database system supports multiple connections and multiple statements per connection.

Unicode support

The DB2 driver supports Unicode data types if the database was created with a multi-byte character set.

The driver maps the following DB2 data types to Unicode data types:

DB2 Data Type	Mapped to . . .
Graphic	SQL_WCHAR
Long Vargraphic	SQL_WLONGVARIABLE
Vargraphic	SQL_WVARIABLE

This driver supports the Unicode ODBC W (Wide) function calls, such as SQLConnectW. This allows the Driver Manager to transmit these calls directly to the driver. Otherwise, the Driver Manager would incur the additional overhead of converting the W calls to ANSI function calls, and vice versa.

Default Unicode mapping

The default Unicode mapping for an application's `SQL_C_WCHAR` variable is:

Platform	Default Unicode Mapping
Windows	UTF-16
AIX	UTF-8
HP-UX	UTF-8
Solaris	UTF-8
Linux	UTF-8

APPENDIX A

ODBC Functions

About this appendix

This appendix lists the ODBC functions that you can use in SQL statements.

Contents

Topic:	page
ODBC functions	70

ODBC functions

This section lists the scalar functions that ODBC supports. Your database system may not support all of these functions. See the documentation for your database system to find out which functions are supported.

You can use these functions in SQL statements using the following syntax:

```
{fn scalar-function}
```

where scalar-function is one of the functions listed in the following tables. For example:

```
SELECT {fn UCASE(NAME)} FROM EMP
```

String functions

The following table lists the string functions that ODBC supports.

The string functions listed can take the following arguments:

- ◆ `string_exp` can be the name of a column, a string literal, or the result of another scalar function, where the underlying data type is `SQL_CHAR`, `SQL_VARCHAR`, or `SQL_LONGVARCHAR`.
- ◆ `start`, `length`, and `count` can be the result of another scalar function or a literal numeric value, where the underlying data type is `SQL_TINYINT`, `SQL_SMALLINT`, or `SQL_INTEGER`.

The string functions are one-based; that is, the first character in the string is character 1.

Character string literals must be surrounded in single quotation marks.

Function	Returns
<code>ASCII(string_exp)</code>	ASCII code value of the leftmost character of <code>string_exp</code> as an integer.
<code>BIT_LENGTH(string_exp)</code>	The length in bits of the string expression.
<code>CHAR(code)</code>	The character with the ASCII code value specified by <code>code</code> . <code>code</code> should be between 0 and 255; otherwise, the return value is data-source dependent.

Function	Returns
CHAR_LENGTH(<i>string_exp</i>)	The length in characters of the string expression, if the string expression is of a character data type; otherwise, the length in bytes of the string expression (the smallest integer not less than the number of bits divided by 8). (This function is the same as the CHARACTER_LENGTH function.)
CHARACTER_LENGTH(<i>string_exp</i>)	The length in characters of the string expression if the string expression is of a character data type; otherwise, the length in bytes of the string expression (the smallest integer not less than the number of bits divided by 8). (This function is the same as the CHAR_LENGTH function.)
CONCAT(<i>string_exp1</i> , <i>string_exp2</i>)	The string resulting from concatenating <i>string_exp2</i> and <i>string_exp1</i> . The string is system dependent.
DIFFERENCE(<i>string_exp1</i> , <i>string_exp2</i>)	An integer value that indicates the difference between the values returned by the SOUNDEX function for <i>string_exp1</i> and <i>string_exp2</i> .
INSERT(<i>string_exp1</i> , <i>start</i> , <i>length</i> , <i>string_exp2</i>)	A string where <i>length</i> characters have been deleted from <i>string_exp1</i> beginning at <i>start</i> and where <i>string_exp2</i> has been inserted into <i>string_exp</i> , beginning at <i>start</i> .
LCASE(<i>string_exp</i>)	Uppercase characters in <i>string_exp</i> converted to lowercase.
LEFT(<i>string_exp</i> , <i>count</i>)	The count of characters of <i>string_exp</i> .
LENGTH(<i>string_exp</i>)	The number of characters in <i>string_exp</i> , excluding trailing blanks and the string termination character.

Function	Returns
LOCATE(<i>string_exp1</i> , <i>string_exp2</i> [, <i>start</i>])	The starting position of the first occurrence of <i>string_exp1</i> within <i>string_exp2</i> . If <i>start</i> is not specified, the search begins with the first character position in <i>string_exp2</i> . If <i>start</i> is specified, the search begins with the character position indicated by the value of <i>start</i> . The first character position in <i>string_exp2</i> is indicated by the value 1. If <i>string_exp1</i> is not found, 0 is returned.
LTRIM(<i>string_exp</i>)	The characters of <i>string_exp</i> , with leading blanks removed.
OCTET_LENGTH(<i>string_exp</i>)	The length in bytes of the string expression. The result is the smallest integer not less than the number of bits divided by 8.
POSITION(<i>character_exp</i> IN <i>character_exp</i>)	The position of the first character expression in the second character expression. The result is an exact numeric with an implementation-defined precision and a scale of 0.
REPEAT(<i>string_exp</i> , <i>count</i>)	A string composed of <i>string_exp</i> repeated <i>count</i> times.
REPLACE(<i>string_exp1</i> , <i>string_exp2</i> , <i>string_exp3</i>)	Replaces all occurrences of <i>string_exp2</i> in <i>string_exp1</i> with <i>string_exp3</i> .
RIGHT(<i>string_exp</i> , <i>count</i>)	The rightmost count of characters in <i>string_exp</i> .
RTRIM(<i>string_exp</i>)	The characters of <i>string_exp</i> with trailing blanks removed.
SOUNDEX(<i>string_exp</i>)	A data-source-dependent string representing the sound of the words in <i>string_exp</i> .
SPACE(<i>count</i>)	A string consisting of <i>count</i> spaces.
SUBSTRING(<i>string_exp</i> , <i>start</i> , <i>length</i>)	A string derived from <i>string_exp</i> beginning at the character position <i>start</i> for <i>length</i> characters.
UCASE(<i>string_exp</i>)	Lowercase characters in <i>string_exp</i> converted to uppercase.

Numeric functions

The following table lists the numeric functions that ODBC supports.

The numeric functions listed can take the following arguments:

- ◆ **numeric_exp** can be a column name, a numeric literal, or the result of another scalar function, where the underlying data type is SQL_NUMERIC, SQL_DECIMAL, SQL_TINYINT, SQL_SMALLINT, SQL_INTEGER, SQL_BIGINT, SQL_FLOAT, SQL_REAL, or SQL_DOUBLE.
- ◆ **float_exp** can be a column name, a numeric literal, or the result of another scalar function, where the underlying data type is SQL_FLOAT.
- ◆ **integer_exp** can be a column name, a numeric literal, or the result of another scalar function, where the underlying data type is SQL_TINYINT, SQL_SMALLINT, SQL_INTEGER, or SQL_BIGINT.

Function	Returns
ABS(<i>numeric_exp</i>)	Absolute value of <i>numeric_exp</i> .
ACOS(<i>float_exp</i>)	Arc-cosine of <i>float_exp</i> as an angle in radians.
ASIN(<i>float_exp</i>)	Arc-sine of <i>float_exp</i> as an angle in radians.
ATAN(<i>float_exp</i>)	Arc-tangent of <i>float_exp</i> as an angle in radians.
ATAN2(<i>float_exp1</i> , <i>float_exp2</i>)	Arc-tangent of the x and y coordinates, specified by <i>float_exp1</i> and <i>float_exp2</i> as an angle in radians.
CEILING(<i>numeric_exp</i>)	Smallest integer greater than or equal to <i>numeric_exp</i> .
COS(<i>float_exp</i>)	Cosine of <i>float_exp</i> as an angle in radians.
COT(<i>float_exp</i>)	Cotangent of <i>float_exp</i> as an angle in radians.
DEGREES(<i>numeric_exp</i>)	Number if degrees converted from <i>numeric_exp</i> radians.
EXP(<i>float_exp</i>)	Exponential value of <i>float_exp</i> .

Function	Returns
FLOOR(<i>numeric_exp</i>)	Largest integer less than or equal to <i>numeric_exp</i> .
LOG(<i>float_exp</i>)	Natural log of <i>float_exp</i> .
LOG10(<i>float_exp</i>)	Base 10 log of <i>float_exp</i> .
MOD(<i>integer_exp1</i> , <i>integer_exp2</i>)	Remainder of <i>integer_exp1</i> divided by <i>integer_exp2</i> .
PI()	Constant value of pi as a floating-point number.
POWER(<i>numeric_exp</i> , <i>integer_exp</i>)	Value of <i>numeric_exp</i> to the power of <i>integer_exp</i> .
RADIANS(<i>numeric_exp</i>)	Number of radians converted from <i>numeric_exp</i> degrees.
RAND([<i>integer_exp</i>])	Random floating-point value using <i>integer_exp</i> as the optional seed value.
ROUND(<i>numeric_exp</i> , <i>integer_exp</i>)	<i>numeric_exp</i> rounded to <i>integer_exp</i> places right of the decimal (left of the decimal if <i>integer_exp</i> is negative).
SIGN(<i>numeric_exp</i>)	Indicator of the sign of <i>numeric_exp</i> . If <i>numeric_exp</i> < 0, -1 is returned. If <i>numeric_exp</i> = 0, 0 is returned. If <i>numeric_exp</i> > 0, 1 is returned.
SIN(<i>float_exp</i>)	Sine of <i>float_exp</i> , where <i>float_exp</i> is an angle in radians.
SQRT(<i>float_exp</i>)	Square root of <i>float_exp</i> .
TAN(<i>float_exp</i>)	Tangent of <i>float_exp</i> , where <i>float_exp</i> is an angle in radians.
TRUNCATE(<i>numeric_exp</i> , <i>integer_exp</i>)	<i>numeric_exp</i> truncated to <i>integer_exp</i> places right of the decimal. (If <i>integer_exp</i> is negative, truncation is to the left of the decimal).

Date and Time Functions

The following table lists the date and time functions that ODBC supports.

The date and time functions listed can take the following arguments:

- ◆ `date_exp` can be a column name, a date or timestamp literal, or the result of another scalar function, where the underlying data type can be represented as `SQL_CHAR`, `SQL_VARCHAR`, `SQL_DATE`, or `SQL_TIMESTAMP`.
- ◆ `time_exp` can be a column name, a timestamp or timestamp literal, or the result of another scalar function, where the underlying data type can be represented as `SQL_CHAR`, `SQL_VARCHAR`, `SQL_TIME`, or `SQL_TIMESTAMP`.
- ◆ `timestamp_exp` can be a column name, a time, date, or timestamp literal; or the result of another scalar function, where the underlying data type can be represented as `SQL_CHAR`, `SQL_VARCHAR`, `SQL_TIME`, `SQL_DATE`, or `SQL_TIMESTAMP`.

Function	Returns
<code>CURRENT_DATE()</code>	Current date.
<code>CURRENT_TIME[(time-precision)]</code>	Current local time. The <i>time-precision</i> argument determines the seconds precision of the returned value.
<code>CURRENT_TIMESTAMP[(timestamp-precision)]</code>	Current local date and local time as a timestamp value. The <i>timestamp-precision</i> argument determines the seconds precision of the returned timestamp.
<code>CURDATE()</code>	Current date as a date value.
<code>CURTIME()</code>	Current local time as a time value.
<code>DAYNAME(date_exp)</code>	Character string containing a data-source-specific name of the day for the day portion of <i>date_exp</i> .
<code>DAYOFMONTH(date_exp)</code>	Day of the month in <i>date_exp</i> as an integer value.
<code>DAYOFWEEK(date_exp)</code>	Day of the week in <i>date_exp</i> as an integer value.
<code>DAYOFYEAR(date_exp)</code>	Day of the year in <i>date_exp</i> as an integer value.
<code>HOUR(time_exp)</code>	Hour in <i>time_exp</i> as an integer value.
<code>MINUTE(time_exp)</code>	Minute in <i>time_exp</i> as an integer value.

Function	Returns
MONTH(<i>date_exp</i>)	Month in <i>date_exp</i> as an integer value.
MONTHNAME(<i>date_exp</i>)	Character string containing the data source-specific name of the month.
NOW()	Current date and time as a timestamp value.
QUARTER(<i>date_exp</i>)	Quarter in <i>date_exp</i> as an integer value.
SECOND(<i>time_exp</i>)	Second in <i>time_exp</i> as an integer value.
TIMESTAMPADD(<i>interval</i> , <i>integer_exp</i> , <i>time_exp</i>)	Timestamp calculated by adding <i>integer_exp</i> intervals of type <i>interval</i> to <i>time_exp</i> . The interval can be SQL_TSI_FRAC_SECOND, SQL_TSI_SECOND, SQL_TSI_MINUTE, SQL_TSI_HOUR, SQL_TSI_DAY, SQL_TSI_WEEK, SQL_TSI_MONTH, SQL_TSI_QUARTER, or SQL_TSI_YEAR Fractional seconds are expressed in billionths of a second.
TIMESTAMPDIFF(<i>interval</i> , <i>time_exp1</i> , <i>time_exp2</i>)	Integer number of intervals of type <i>interval</i> by which <i>time_exp2</i> is greater than <i>time_exp1</i> . <i>interval</i> has the same values as TIMESTAMPADD. Fractional seconds are expressed in billionths of a second.
WEEK(<i>date_exp</i>)	Week of the year in <i>date_exp</i> as an integer value.
YEAR(<i>date_exp</i>)	Year in <i>date_exp</i> . The range is data-source dependent.

System functions

The following table lists the system functions that ODBC supports.

Function	Returns
DATABASE()	Name of the database, corresponding to the connection handle (hdbc).
IFNULL(<i>exp</i> , <i>value</i>)	<i>value</i> , if <i>exp</i> is null.
USER()	Authorization name of the user.

APPENDIX B

Values for IANAAppCodePage Connection String Attribute

About this appendix This appendix contains specific information about using Connect ODBC in the UNIX environments.

Contents

Topic:	page
Table of IANAAppCodePage values	78

Table of IANAAppCodePage values

The following table lists frequently-used valid values for the IANAAppCodePage connection string attribute. See the appropriate individual driver chapter for information about this attribute. A complete list of values and aliases is available at <http://www.iana.org/assignments/character-sets>.

NOTE: The IANAAppCodePage connection string attribute replaces the AppCodePage connection string attribute in earlier versions of the Connect for ODBC drivers. The drivers are backwardly compatible with the AppCodePage attribute, but only the IANAAppCodePage attribute should now be used.

Value	Description
3	US_ASCII
4	ISO_8859_1
5	ISO_8859_2
6	ISO_8859_3
7	ISO_8859_4
8	ISO_8859_5
9	ISO_8859_6
10	ISO_8859_7
11	ISO_8859_8
12	ISO_8859_9
13	ISO_8859_10
16	JIS_Encoding
17	Shift_JIS
18	EUC_JP
30	ISO_646_IRV
36	KSC_5601
37	ISO_2022_KR
38	EUC_KR
39	ISO_2022_JP

Appendix B. Values for IANAAppCodePage Connection String Attribute

Value	Description
40	ISO_2022_JP_2
57	GB_2312_80
104	ISO_2022_CN
105	ISO_2022_CN_EXT
109	ISO_8859_13
110	ISO_8859_14
111	ISO_8859_15
113	GBK
2004	HP_ROMAN8
2009	IBM850
2010	IBM852
2011	IBM437
2013	IBM862
2024	Windows_31J
2025	JB2312
2026	Big5
2027	macintosh
2028	IBM037
2030	IBM273
2033	IBM277
2034	IBM278
2035	IBM280
2037	IBM284
2038	IBM285
2039	IBM290
2040	IBM297
2041	IBM420
2043	IBM424

Value	Description
2044	IBM500
2045	IBM851
2046	IBM855
2047	IBM857
2048	IBM860
2049	IBM861
2050	IBM863
2051	IBM864
2052	IBM865
2053	IBM424
2054	IBM869
2055	IBM870
2056	IBM871
2062	IBM918
2063	IBM1026
2084	KO18_R
2085	HZ_GB_2312
2086	IBM866
2087	IBM775
2089	IBM0858
2091	IBM1140
2092	IBM1141
2093	IBM1142
2094	IBM1143
2095	IBM1144
2096	IBM1145
2097	IBM1146
2098	IBM1147

Appendix B. Values for IANAAppCodePage Connection String Attribute

Value	Description
2099	IBM1148
2100	IBM1149
2102	IBM1047
2250	WINDOWS_1250
2251	WINDOWS_1251
2252	WINDOWS_1252
2253	WINDOWS_1253
2254	WINDOWS_1254
2255	WINDOWS_1255
2256	WINDOWS_1256
2257	WINDOWS_1257
2258	WINDOWS_1258
2259	TIS_620



Index

A

about iAnywhere Solutions ODBC drivers	2	ASE driver	18
ACP connection parameter		Oracle driver	38
ASE driver	18	AS connection parameter	
DB2 driver	60	ASE driver	18
Oracle driver	38	Oracle driver	38
Adaptive Server Enterprise driver	11	ASCT connection parameter	
Adaptive Server Enterprise driver		DB2 driver	60
about	11	AUT connection parameter	
BLOB data maximum size	13	ASE driver	18
code pages	18	Oracle driver	38
connecting using a connection string	18		
connecting using a logon dialog box	17	B	
connection string attributes	18	bind packages	
connections supported	30	DB2 driver	53
data source configuration	15	bind19	
data types	27	iAnywhere Solutions UNIX ODBC driver for	
driver requirements	12	DB2	53
key ODBC settings	13	BLOBs	
query timeout	30	ASE driver maximum size	13
statements supported	30	Oracle driver maximum size	33
AddStringToCreateTable connection parameter		C	
DB2 driver	60	CachedCursorLimit connection parameter	
AID connection parameter		Oracle driver	38
DB2 driver	60	CachedDescriptionLimit connection parameter	
AIX		Oracle driver	38
ODBC drivers	4	CatalogIncludesSynonyms connection parameter	
AlternateID connection parameter		Oracle driver	38
DB2 driver	60	CatalogOptions connection parameter	
APP connection parameter		Oracle driver	38
ASE driver	18	CCL connection parameter	
AppCodePage connection parameter		Oracle driver	38
ASE driver	18	CCS connection parameter	
DB2 driver	60	ASE driver	18
Oracle driver	38	CDL connection parameter	
ApplicationName connection parameter		Oracle driver	38
ASE driver	18	CharSet connection parameter	
ApplicationUsingThreads connection parameter		ASE driver	18
ASE driver	18	choosing drivers	
Oracle driver	38	using iAnywhere Solutions ODBC drivers	1
ArraySize connection parameter		CIS connection parameter	
		Oracle driver	38

CO connection parameter	
Oracle driver	38
code pages	
ASE driver	18
connection string attributes for iAnywhere	
Solutions ODBC drivers	77
DB2 driver	60
Oracle driver	38
COL connection parameter	
DB2 driver	60
Collection connection parameter	
DB2 driver	60
configuring	
ODBC drivers on UNIX	4
ODBC drivers on Windows	3
configuring data sources	
ASE driver	15
DB2 driver	54
iAnywhere Solutions Oracle driver	34
Connect ODBC	
Sybase Wire Protocol	77
connecting to a data source using a connection string	
ASE driver	18
DB2 driver	60
Oracle driver	38
connecting to a data source using a logon dialog box	
ASE driver	17
DB2 driver	59
Oracle driver	37
connection parameters	
ASE driver	18
DB2 driver	60
Oracle driver	38
connection string attributes	
ASE driver	18
DB2	60
Oracle	38
connections supported	
DB2 driver	66
Oracle driver	49
Sybase	30
conventions	
documentation	viii
CS connection parameter	
ASE driver	18
CursorCacheSize connection parameter	
ASE driver	18

D

DAP connection parameter	
Oracle driver	38
data sources	
configuring ASE	15
configuring DB2	54
configuring for Oracle driver	34
connecting to ASE via a connection string	18
connecting to ASE via a logon dialog box	17
connecting via connection string in DB2	60
connecting via connection string with Oracle	38
connecting via logon dialog box for DB2	59
connecting via logon dialog box for Oracle	37
data types	
DB2	65
Oracle Wire Protocol	45
Sybase	27
Database connection parameter	
ASE driver	18
DB2 driver	60
DataSourceName connection parameter	
ASE driver	18
DB2 driver	60
Oracle driver	38
date and time functions	
iAnywhere Solutions ODBC drivers	74
DB connection parameter	
ASE driver	18
DB2 driver	60
DB2	<i>see also</i> IBM DB2
driver	51
DB2 driver	
about	51
code pages	60
connecting using a connection string	60
connecting using a logon dialog box	59
connection string attributes	60
connections supported	66
data source configuration	54
data types	65
driver requirements	52
isolation levels	66
key ODBC settings	53
locking levels	66
statements supported	66
stored procedure support	66
supported platforms	52

Unicode support	67	ASE driver	18
DecimalPrecision connection parameter		DynamicSections connection parameter	
DB2 driver key setting	53	DB2 driver	60
default Unicode mapping		E	
Oracle driver	47	EDP connection parameter	
DefaultLongDataBuffLen connection parameter		ASE driver	18
Adaptive Server Enterprise driver key setting	13	Oracle driver	38
ASE driver	18	EnableDescribeParam connection parameter	
Oracle driver	38	ASE driver	18
Oracle driver key setting	33	Oracle driver	38
DescribeAtPrepare connection parameter		EnableQuotedIdentifiers connection parameter	
Oracle driver	38	ASE driver	18
DistributedTransactionModel connection parameter		EnableScrollableCursors connection parameter	
ASE driver	18	Oracle driver	38
DLDBL connection parameter		EnableStaticCursorsForLongData connection	
Adaptive Server Enterprise driver key setting	13	parameter	
ASE driver	18	Oracle driver	38
Oracle driver	38	EnableTimestampWithTimezone connection	
Oracle driver key setting	33	parameter	
documentation		Oracle driver	38
conventions	viii	EQI connection parameter	
SQL Anywhere Studio	vi	ASE driver	18
DP connection parameter		error messages	
DB2 driver key setting	53	iAnywhere Solutions ODBC drivers	9
driver file names		iAnywhere Solutions ODBC drivers for UNIX	10
iAnywhere Solutions drivers on UNIX	7	ESC connection parameter	
iAnywhere Solutions drivers on Windows	3	Oracle driver	38
driver requirements		ESCLD connection parameter	
ASE driver	12	Oracle driver	38
DB2 driver	52	ETWT connection parameter	
Oracle driver	32	Oracle driver	38
drivers		F	
about iAnywhere Solutions ODBC drivers	1	failover	
ASE	11	ASE driver	18
DB2	51	FailoverNetworkAddress connection parameter	
Oracle	31	ASE driver	18
DS connection parameter		feedback	
DB2 driver	60	documentation	ix
DSN		providing	ix
configuring ASE	15	FNA connection parameter	
configuring DB2	54	ASE driver	18
configuring for Oracle driver	34	G	
DSN connection parameter		GA connection parameter	
ASE driver	18		
DB2 driver	60		
Oracle driver	38		
DTM connection parameter			

DB2 driver	60	code pages	60
GE connection parameter		connecting using a connection string	60
DB2 driver	60	connecting using a logon dialog box	59
GrantAuthid connection parameter		connection string attributes	60
DB2 driver	60	data source configuration	54
GrantExecute connection parameter		data types	65
DB2 driver	60	driver requirements	52
		key ODBC settings	53
		stored procedure support	66
		Unicode support	67
H		IF connection parameter	
high availability		ASE driver	18
ASE driver	18	IFSN connection parameter	
HOST connection parameter		ASE driver	18
Oracle driver	38	IL connection parameter	
HostName connection parameter		DB2 driver	60
Oracle driver	38	InitializationString connection parameter	
		ASE driver	18
I		InstallDir	
IACP connection parameter		odbc.ini file	4
ASE driver	18	InterfacesFile connection parameter	
DB2 driver	60	ASE driver	18
Oracle driver	38	InterfacesFileServerName connection parameter	
IANAAppCodePage connection parameter		ASE driver	18
ASE driver	18	introduction to iAnywhere Solutions ODBC drivers	1
DB2 driver	60	introduction to ODBC drivers	
Oracle driver	38	MobiLink and remote data access	1
IANAAppCodePage connection string attributes		IP connection parameter	
for iAnywhere Solutions ODBC drivers	77	DB2 driver	60
iAnywhere Solutions ODBC driver		IPAddress connection parameter	
Oracle	31	DB2 driver	60
iAnywhere Solutions ODBC driver for DB2	51	IS connection parameter	
iAnywhere Solutions ODBC driver for Oracle	31	ASE driver	18
iAnywhere Solutions ODBC driver for Sybase		isolation and lock levels supported	
Adaptive Server Enterprise	11	DB2 driver	66
iAnywhere Solutions ODBC drivers		Oracle driver	49
about	1	isolation levels	
ASE	11	DB2 driver	66
driver names on UNIX	7	Oracle driver	49
driver names on Windows	3	IsolationLevel connection parameter	
error messages	9	DB2 driver	60
functions	69		
IANAAppCodePage connection string	77		
UNIX	4		
Windows	3		
IBM DB2		K	
driver	51	key settings for MobiLink and Adaptive Server	
IBM DB2 driver		Enterprise	
about	51	iAnywhere Solutions ODBC drivers	13
		key settings for MobiLink and DB2	

iAnywhere Solutions ODBC drivers	53	NetworkLibraryName connection parameter	
key settings for MobiLink and Oracle		ASE driver	18
iAnywhere Solutions ODBC drivers	33	newsgroups	
L		technical support	ix
LANG connection parameter		NLN connection parameter	
ASE driver	18	ASE driver	18
Language connection parameter		number of connections and statements supported	
ASE driver	18	ASE driver	30
LD_LIBRARY_PATH environment variable		DB2 driver	66
iAnywhere Solutions ODBC drivers	8	Oracle driver	49
library path environment variable		numeric functions	
iAnywhere Solutions ODBC drivers	8	iAnywhere Solutions ODBC drivers	73
Linux		O	
iAnywhere Solutions ODBC driver environment		ODBC	
variable required for Redhat 9	8	ASE driver	11
ODBC drivers	4	DB2 driver	51
LOC connection parameter		iAnywhere Solutions drivers	1
DB2 driver	60	Oracle driver	31
LocalTimezoneOffset connection parameter		scalar functions for iAnywhere Solutions ODBC	
Oracle driver	38	drivers	70
Location connection parameter		ODBC conformance level	
DB2 driver	60	Oracle driver	49
locking levels		ODBC drivers	
DB2 driver	66	about iAnywhere Solutions ODBC drivers	1
Oracle driver	49	AIX	4
LockTimeout connection parameter		ASE	11
Oracle driver	38	DB2	51
LogonID connection parameter		Linux	4
ASE driver	18	Oracle	31
DB2 driver	60	Solaris	4
Oracle driver	38	ODBC functions	
LTO connection parameter		iAnywhere Solutions ODBC drivers	70
Oracle driver	38	MobiLink and remote access	69
LTZO connection parameter		odbc.ini	
Oracle driver	38	ODBC configuration file for UNIX	4
M		ODBCINI environment variable	7, 8
MobiLink		OP connection parameter	
ODBC drivers	1	ASE driver	18
N		OptimizePrepare connection parameter	
NA connection parameter		Adaptive Server Enterprise driver key setting	13
ASE driver	18	ASE driver	18
NetworkAddress connection parameter		Oracle	
ASE driver	18	driver	31
		Oracle driver	
		about	31
		BLOB data maximum size	33

code pages	38
connecting using a connection string	38
connecting using a logon dialog box	37
connection string attributes	38
connections supported	49
data source configuration	34
data types	45
default Unicode mapping	47
driver requirements	32
isolation levels	49
key ODBC settings	33
locking levels	49
statements supported	49
Unicode mapping	47
Unicode support	46

P

Package connection parameter	
DB2 driver	60
PackageOwner connection parameter	
DB2 driver	60
PacketSize connection parameter	
ASE driver	18
Password connection parameter	
ASE driver	18
DB2 driver	60
Oracle driver	38
PCK connection parameter	
DB2 driver	60
PO connection parameter	
DB2 driver	60
PORT connection parameter	
DB2 driver	60
Oracle driver	38
PortNumber connection parameter	
Oracle driver	38
ProcedureRetResults connection parameter	
Oracle driver	38
Oracle driver usage	48
PRR connection parameter	
Oracle driver	38
PS connection parameter	
ASE driver	18
PWD connection parameter	
ASE driver	18
DB2 driver	60
Oracle driver	38

Q

QUERY_TIMEOUT	
setting for the ASE driver	30
quick start for UNIX	
iAnywhere Solutions ODBC drivers	4
quick start for Windows	
iAnywhere Solutions ODBC drivers	3

R

RaiseErrorPositionBehavior connection parameter	
ASE driver	18
remote data access	
ODBC drivers	1
REPB connection parameter	
ASE driver	18

S

scalar functions	
iAnywhere Solutions ODBC drivers	70
SelectMethod attribute	
ASE driver usage	30
SelectMethod connection parameter	
Adaptive Server Enterprise driver key setting	13
ASE driver	18
setting the library path environment variable	
iAnywhere Solutions ODBC drivers	8
SID connection parameter	
Oracle driver	38
Oracle driver key setting	33
SM connection parameter	
ASE driver	18
Solaris	
ODBC drivers	4
SQL Anywhere Studio	
documentation	vi
starting the ODBC administrator	
iAnywhere Solutions ODBC drivers for Windows	3
statements supported	
ASE driver	30
DB2 driver	66
Oracle driver	49
stored procedure results	
Oracle driver	48
stored procedure support	
DB2 driver	66

stored procedures		environment ODBCINI	8
DB2 driver	66	environment system information file (.odbc.ini)	4
string functions		error messages	10
iAnywhere Solutions ODBC drivers	70	UNIX drivers	
support		ASE	11
newsgroups	ix	DB2	51
support for query timeout		Oracle	31
ASE driver	30	UNIX error handling	
supported features		iAnywhere Solutions ODBC drivers	10
ASE driver	30	UseCurrentSchema connection parameter	
DB2 driver	66	Oracle driver	38
Oracle driver	49		
system functions		V	
iAnywhere Solutions ODBC driver support	76	values for IANAAppCodePage connection string attribute	
system information file (.odbc.ini)	4	iAnywhere Solutions ODBC drivers	77
T			
table of IANAAppCodePage values	78	W	
TCDT connection parameter		WH connection parameter	
ASE driver	18	DB2 driver	60
TcpPort connection parameter		DB2 driver key setting	53
DB2 driver	60	Windows drivers	
technical support		ASE	11
newsgroups	ix	DB2	51
TightlyCoupledDistributedTransactions connection parameter		wire protocol drivers	
ASE driver	18	ASE	11
time functions		DB2	51
iAnywhere Solutions ODBC drivers	74	Oracle	31
U		WithHold connection parameter	
UCS connection parameter		DB2 driver	60
Oracle driver	38	DB2 driver key setting	53
UDB	<i>see also</i> IBM DB2	WKID connection parameter	
UID connection parameter		ASE driver	18
ASE driver	18	WorkStationID connection parameter	
DB2 driver	60	ASE driver	18
Oracle driver	38	wq	
Unicode mapping		prefix for iAnywhere Solutions drivers	3
Oracle driver	47	X	
Unicode support		XAOpenStringParameters connection parameter	
DB2 driver	67	ASE driver	18
Oracle driver	46	XAOSP connection parameter	
UNIX		ASE driver	18
code page connection string attributes for			
iAnywhere Solutions ODBC drivers	77		
environment library path	8		