

Oracle® Access Manager for AS/400

Installation and User's Guide

10g Release 2 (10.2) for IBM iSeries OS/400

B16223-02

August 2007

Oracle Access Manager for AS/400 Installation and User's Guide, 10g Release 2 (10.2) for IBM iSeries OS/400
B16223-02

Copyright © 2002, 2007, Oracle. All rights reserved.

Primary Author: Maitreyee Chaliha

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Preface	vii
Intended Audience.....	vii
Documentation Accessibility	vii
Product Name.....	viii
Platform Name	viii
Related Documents	viii
Conventions	ix
Storage Measurements	ix
1 Introduction	
1.1 Overview	1-1
1.2 Architecture of Oracle Access Manager for AS/400.....	1-2
1.2.1 Components	1-3
1.3 Features	1-4
1.3.1 Powerful Oracle SQL Language	1-4
1.3.2 Leverage Existing Applications.....	1-4
1.3.3 Universal Access.....	1-4
1.3.4 Data Type Conversions	1-4
1.3.5 Support for Distributed Applications.....	1-5
1.4 Calling Oracle Access Manager for AS/400	1-5
1.5 IBM Distributed Relational Databases	1-5
1.6 Activation Groups	1-6
2 Release Information	
2.1 Product Set	2-1
2.2 Changes and Enhancements	2-1
2.2.1 Release 10.2.0.1.0.....	2-1
2.2.2 Bugs Fixed for Release 10.2.0.1.0	2-1
2.3 Known Problems and Restrictions	2-2
2.3.1 Known Problems	2-2
2.3.2 Known Restrictions	2-3
3 System Requirements	
3.1 Hardware Requirements.....	3-1
3.1.1 Processor	3-1

3.1.2	CD-ROM Drive	3-1
3.1.3	Disk Space.....	3-1
3.2	Software Requirements	3-2
3.2.1	Operating System	3-2
3.2.2	Communications.....	3-2
3.2.3	Oracle Server	3-2
3.2.4	PTF Requirements	3-2
3.2.5	Additional Software	3-2
3.3	Recommended Documentation	3-2

4 Installing Access Manager

4.1	Checklists	4-1
4.1.1	Installation Checklist.....	4-1
4.1.2	Post-Installation Checklist.....	4-1
4.2	Installation	4-2
4.2.1	Steps for Installation	4-2
4.2.2	Step 1: Mount the Product CD-ROM	4-2
4.2.3	Step 2: Start the Installation Process	4-2
4.2.4	Step 3: Select the Installation Library Name.....	4-2
4.2.5	Step 4: Verify Installation Parameters.....	4-3
4.2.6	Step 5: Finish the Installation	4-3
4.3	Post-Installation.....	4-4
4.3.1	Steps for Post-Installation	4-4
4.3.2	Step 1: Add Oracle Access Installation Library to Library List	4-4
4.3.3	Step 2: Change Password for User Profile	4-4
4.3.4	Step 3: Configure Oracle Net	4-5
4.3.5	Step 4: Test Connectivity to Oracle Server	4-5
4.3.6	Step 5: Configure Access Manager.....	4-5
4.3.7	Step 6: Configure Two-Phase Commit (Optional)	4-5
4.3.8	Step 7: Retest Connectivity to Oracle Server (Optional)	4-5

5 Oracle Net

5.1	Oracle Net File Name Structure.....	5-1
5.2	Connecting to Oracle through Oracle Net.....	5-2
5.3	Defining the Oracle Server to Oracle Access Manager	5-2
5.3.1	Remote Database Directory Entry	5-3
5.3.2	ORA(TNSNAMES).....	5-3
5.3.3	Configuration Entries.....	5-3
5.4	Oracle Net for TCP/IP AS/400 Configuration.....	5-4
5.4.1	Steps for TCP/IP AS/400 Configuration	5-4
5.4.2	Step 1: Create Relational Database Directory Entry	5-4
5.4.3	Step 2: Define TNSNAMES Entry	5-5

6 Configuring Access Manager

6.1	Access Manager Commands	6-1
6.2	CHGSQLDFLT	6-2

6.3	CHGSQLNET	6-5
6.4	CHGSQLDBG	6-6
6.5	CRTORAPKG	6-7
6.6	CHG2PCPRM	6-9

7 Distributed Transaction Coordination

7.1	Two-Phase Commit Overview	7-1
7.1.1	Terminology	7-1
7.2	Two-Phase Commit and Oracle Access Manager for AS/400	7-2
7.3	Two-Phase Commit Process	7-2
7.3.1	Phase 1: Prepare	7-3
7.3.2	Phase 2: Commit	7-3
7.4	In-Doubt Transactions	7-3
7.4.1	Resolving In-Doubt Transactions	7-3
7.5	Enabling Two-Phase Commit Processing	7-4
7.5.1	Checklist for Enabling Two-Phase Commit	7-4
7.5.2	Step 1: Change Access Manager Two-Phase Commit Default	7-5
7.5.3	Step 2: Change the AM400_RID Options	7-5
7.5.4	Step 3: Add or Remove AM400_COM Privileged User IDs	7-7
7.5.5	Step 4: Start the Subsystem that Runs the AM400_RID Program	7-8
7.5.6	Step 5: Start the AM400_COM Program (Optional)	7-8
7.6	Oracle Required Privileges	7-8
7.7	AM400_COM Program	7-9
7.7.1	AM400_COM Commands	7-9

8 Using Access Manager

8.1	Connecting to the Oracle Server	8-1
8.1.1	Connecting through the STRSQL (Interactive SQL) Commands	8-1
8.1.2	Connecting from an AS/400 Precompiled Application	8-2
8.2	Connection Management	8-3
8.3	Describing Tables	8-4
8.4	Stored Procedures	8-4
8.4.1	Advantages of Stored Procedures	8-4
8.4.2	Stored Procedure Parameters	8-5
8.4.3	Declaring Procedures	8-5
8.5	Conversion of Oracle Data Types to DB2/400 Data Types	8-6
8.6	Message and Error Code Processing	8-7

9 Error Diagnosis and Reporting

9.1	Trace Files from the Access Manager	9-1
9.2	Trace Files for Oracle Net	9-2
9.3	Contacting Oracle Support Services	9-2
9.4	Error categories	9-2
9.4.1	Documentation Error	9-2
9.4.2	Incorrect Output	9-2
9.4.3	Oracle External Error	9-3

9.4.4	Performance.....	9-3
9.4.5	Missing Functionality.....	9-3
9.5	How to Provide Items for Problem Diagnosis.....	9-3

A Sample Programs

A.1	Sample Access Manager C Program	A-1
A.2	Sample Access Manager COBOL Program	A-3
A.3	Sample Access Manager RPG Program	A-4

B Globalization Support

B.1	Access Manager Globalization Support	B-1
B.2	Oracle Character Set Names.....	B-1
B.3	Oracle NLS_LANG Specifications.....	B-2

Index

Preface

This manual describes how to install, configure, and use the Oracle Access Manager for AS/400 software on the IBM iSeries.

Intended Audience

Read this guide if you are responsible for performing tasks such as:

- installing and configuring the Oracle Access Manager gateway
- administering the Oracle Access Manager gateway
- using the Oracle Access Manager gateway

Understand the fundamentals of both the Oracle Access Manager and OS/400 before using this guide for installation or system administration.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

Product Name

The complete name for this product is Oracle Access Manager for AS/400, and it is also abbreviated in this document as Access Manager, Oracle Access Manager, Oracle Access Manager gateway, or AM4AS/400.

Platform Name

In this manual, the platform is called OS/400. The terms AS/400 and OS/400 both refer to the IBM iSeries computers and operating system, which are referred to by IBM as IBM eServer iSeries servers, and include the IBM i5.

Related Documents

For information about Oracle products, refer to the following manuals in 10g Release 1 (10.1):

- *Oracle Database Administrator's Guide*
- *Oracle Streams Advanced Queuing User's Guide*
- *Oracle Database Advanced Security Administrator's Guide*
- *Oracle Database Backup and Recovery User's Guide*
- *Oracle Database Backup and Recovery Basics*
- *Oracle Database Concepts*
- *Oracle Database Error Messages*
- *Oracle Database Heterogeneous Connectivity Administrator's Guide*
- *Oracle C++ Call Interface Programmer's Guide*
- *Oracle Call Interface Programmer's Guide*
- *Oracle Database High Availability Architecture and Best Practices*
- *Oracle Database Net Services Administrator's Guide*
- *Oracle Database Net Services Reference*
- *Oracle Database New Features Guide*
- *Oracle Secure Backup Reference*
- *Oracle Database Performance Tuning Guide*
- *Oracle Database Backup and Recovery Quick Start Guide*
- *Oracle Database Backup and Recovery Reference*
- *Oracle Database Reference*
- *Oracle Database SQL Language Quick Reference*
- *Oracle Database SQL Language Reference*
- *SQL*Plus Quick Reference*

- *SQL*Plus User's Guide and Reference*
- *Oracle Streams Replication Administrator's Guide*
- *Oracle Streams Concepts and Administration*
- *Oracle Database Upgrade Guide*

For information about IBM products, refer to the IBM documents for your platform and operating system.

Conventions

Examples of input and output to the system are shown in a special font:

```
WRKCFGSTS *DEV device_name
```

Output is always shown as it actually appears. For input, the following conventions apply:

- The use of monospaced uppercase font, such as `WRKCFGSTS`, indicates a word or phrase that must be entered exactly as spelled.
- The use of monospaced Italic font, such as *device_name*, indicates a variable, which is a word or phrase for which you must substitute your own word or phrase, such as an actual device name.
- [] Square brackets indicate that the enclosed arguments are optional. Do not enter the brackets themselves.
- < > Angle brackets enclose user-supplied names. Do not enter the brackets themselves.
- | Vertical lines separate choices.
- ... Ellipses indicate that the preceding item can be repeated. You can enter an arbitrary number of similar items.
- Other punctuation, such as commas and quotes, must be entered as shown unless otherwise specified.

A file ID can appear with both uppercase and lowercase text. The use of *Italic font* indicates that those portions of a file ID that appear in *Italic font* can vary. Reserved words and keywords must always be entered as is. They have reserved meanings within Oracle software products.

Storage Measurements

Storage measurements use the following abbreviations:

- KB for kilobyte, which equals 1024 bytes
- MB for megabyte, which equals 1 048 576 bytes
- GB for gigabyte, which equals 1 073 741 824 bytes

Introduction

Oracle Access Manager for AS/400 enables SQL applications on the IBM iSeries to access and modify data in an Oracle database.

Read this chapter for information about the architecture and features of Oracle Access Manager for AS/400. It includes the following sections:

- [Overview](#) on page 1-1
- [Architecture of Oracle Access Manager for AS/400](#) on page 1-2
- [Features](#) on page 1-4
- [Calling Oracle Access Manager for AS/400](#) on page 1-5
- [IBM Distributed Relational Databases](#) on page 1-5
- [Activation Groups](#) on page 1-6

1.1 Overview

Oracle Database Gateway technology, combined with Oracle Access Manager products, allows you to fully integrate the AS/400 and Oracle systems. The Oracle Database Gateway for DB2/400 and the Oracle Database Gateway for DRDA provide Oracle users with access to AS/400 data as illustrated in [Figure 1-1](#).

Oracle Access Manager provides AS/400 applications with access to Oracle data through SQL statements or stored procedures, as illustrated in [Figure 1-2](#).

Figure 1-1 Oracle Database Gateway

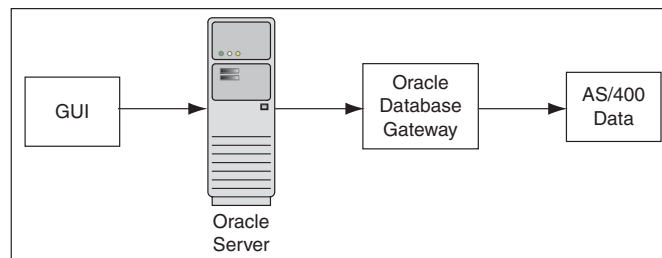
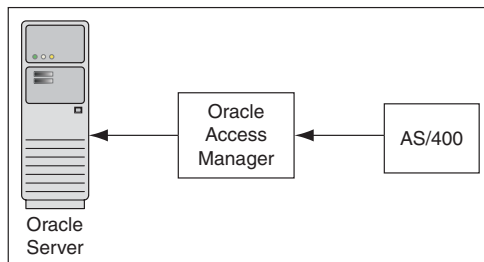


Figure 1–2 Oracle Access Manager

1.2 Architecture of Oracle Access Manager for AS/400

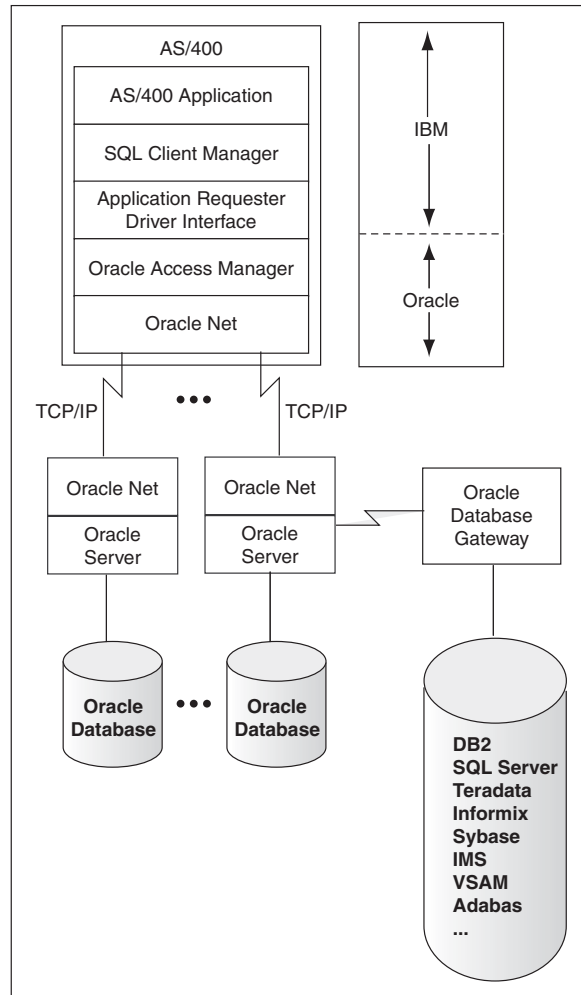
Oracle Access Manager for AS/400 provides OS/400 users with access (through DB2/400) to data that is residing on any supported Oracle platform by the use of Oracle SQL or Oracle stored procedures. Oracle Access Manager connects to the Oracle environment through Oracle Net. The connection from the AS/400 to an Oracle server uses TCP/IP.

Access Manager takes advantage of the IBM Application Requester Driver (ARD) interface to access an Oracle server. For a description of this interface, refer to the IBM document for System API Reference.

Access Manager allows direct access to Oracle data from existing AS/400 applications that are written in C, COBOL, or RPG, or from the AS/400 Interactive Query Manager product (STRSQL). Access Manager intercepts SQL statements during program execution and directs them to the Oracle server.

[Figure 1–3](#) shows the architecture of Oracle Access Manager for AS/400.

Figure 1-3 Oracle Access Manager for AS/400 Architecture



1.2.1 Components

The components interact in the following ways:

Table 1-1 Access Manager Component Interactions

Component	Interaction
AS/400 application	performs processing and invokes embedded SQL to run SQL statements and retrieve results.
SQL Client Manager	manages connections, receives requests from the application, calls the ARD interface to process SQL requests if the connection is to Oracle, and returns results to the application.
ARD Interface	provides an interface between SQL Client Manager and Access Manager.
Access Manager	processes procedure calls, submits SQL requests to the Oracle server, and returns results to SQL Client Manager. It also modifies requests, if necessary, so that they conform to Oracle server syntax.
Oracle server	receives requests from the network and formats them to access Oracle data, then returns results over the network to the client.

Table 1–1 (Cont.) Access Manager Component Interactions

Component	Interaction
Oracle Net	allows databases and their applications to reside on different computers and to communicate as peer applications in a Transparent Networking Substrate (TNS).

1.3 Features

The following section describes the features of the Oracle Access Manager for AS/400

1.3.1 Powerful Oracle SQL Language

AS/400 applications can take advantage of the powerful Oracle SQL language to analyze data that is stored in an Oracle database. Virtually any SQL command can be embedded in an AS/400 application, including `INSERT`, `DELETE`, `UPDATE`, and `SELECT`, as well as a DDL statement. The SQL commands may contain any functions or clauses that are allowed by Oracle SQL. AS/400 applications also can transparently access procedures and packages that are stored in an Oracle database, thereby leveraging business logic that is centralized in the Oracle server. If you have an existing application that uses the IBM `CALL` command to execute an AS/400 procedure, then you can redirect this procedure to call an Oracle PL/SQL stored procedure by using the same IBM `CALL` command.

Note: With two-phase commit enabled, only DML statements (`SELECT`, `INSERT`, `DELETE`, and `UPDATE`) and calls to stored procedures can be used. If two-phase commit is *not* enabled, then any Oracle DDL or DML SQL command may be used.

1.3.2 Leverage Existing Applications

Because Access Manager uses the IBM ARD interface, existing AS/400 applications that are written in C, COBOL, or RPG can directly access Oracle data. Dynamic SQL access to Oracle data is also available through the IBM DB2/400 Interactive Structured Query Language panel (STRSQL). AS/400 applications that are written using ANSI SQL might not require recompilation to access an Oracle server. Access Manager not only allows you to leverage your investment in existing AS/400 products, such as DB2/400 Query Manager and SQL Development Kit, it also provides you the opportunity to redirect AS/400 applications with little or no change to your application. Your application need not write to the proprietary Application Program Interface (API) of a vendor. You embed standard SQL statements in your AS/400 application.

1.3.3 Universal Access

AS/400 applications can now benefit from the fully distributed Oracle architecture. Access Manager allows access to data that is stored in any Oracle Database on your network. When used with Oracle Database Gateway technology, AS/400 applications also have access to over 30 non-Oracle data stores, including DB2/UDB, DB2/400 and other DB2/400 systems, VSAM, Informix, and Sybase.

1.3.4 Data Type Conversions

Access Manager ensures that the appropriate data type conversions are performed between Oracle and AS/400. This includes support for AS/400-specific data types

such as zoned decimal. Access Manager can also translate Oracle date format to the IBM AS/400 default format or to any other date format that you select.

1.3.5 Support for Distributed Applications

Because Access Manager gives the AS/400 application direct access to Oracle data, downloading and uploading large quantities of data to other processors is not necessary. Instead of moving data between computers and thereby risking unsynchronized and inconsistent data, you can access data where it resides. From a single program, the AS/400 application can read AS/400 data and insert it into an Oracle database. You can also read Oracle data and insert it into DB2/400 tables.

1.4 Calling Oracle Access Manager for AS/400

The AS/400 system calls the Oracle Access Manager during the following operations:

- package creation

During package creation, which may be initiated by any of the AS/400 CRTSQLxxx commands (for example, CRTSQLPKG), the AS/400 calls Oracle Access Manager for AS/400 when the RDB parameter matches the RDB name in a relational database directory entry that is associated with the Access Manager. The AS/400 system passes SQL statement information from the user program or package. Oracle Access Manager for AS/400 can use this information to plan SQL access for the program.

However, SQL statements that are passed to Oracle Access Manager for AS/400 are currently ignored when the package is being created. If the connection fails, then the package creation also fails.

- CONNECT processing

Oracle Access Manager for AS/400 is called when a CONNECT statement is issued which specifies an RDB name that matches an RDB name in an AS/400 relational database entry (RDBDIRE) and that entry is associated with the Access Manager. Information is passed to Oracle Access Manager so that it can create an environment in which to run the subsequent SQL statements.

- SQL statement processing

Oracle Access Manager for AS/400 is called during SQL processing when the current connection is to an RDB name in the relational database directory entry that is associated with the Access Manager. The AS/400 passes information about the statement being run so that Oracle Access Manager can process the statement.

1.5 IBM Distributed Relational Databases

DB2/400 provides support for Distributed Relational Database Architecture (DRDA) to allow an application requester to communicate with IBM application servers. In addition, Oracle application servers can be accessed by using the IBM ARD interface. Access Manager uses this interface to integrate Oracle data into your distributed database environment.

DB2/400 supports two levels of distributed databases:

- remote unit of work
- distributed unit of work

A unit of work is sometimes called a transaction. A remote unit of work entails preparing and running SQL statements at only one application server during a unit of work. A distributed unit of work entails preparing and running SQL statements at multiple application servers during a unit of work. However, a single SQL statement can refer only to objects that are located at a single application server.

Oracle Access Manager supports only a distributed unit of work and extends its functionality by allowing a single statement to refer to objects located on more than one database server. For more information about distributed relational databases, refer to the appropriate IBM documentation.

A unit of work is a sequence of SQL commands that the database manager (DB2/400) treats as a single entity. The database manager ensures the consistency of data by verifying that all the data changes that are made during a unit of work are performed, or that none of them are performed. A unit of work is ended by a `COMMIT` or `ROLLBACK` operation.

1.6 Activation Groups

An activation group provides the following:

- run-time data structures to support running programs
- addressing protection
- a logical boundary for message creation
- a logical boundary for application cleanup processing

All SQL connections are managed at the activation group level. The activation group and RDB name are used together to uniquely identify a connection. It is not possible to simultaneously have more than one active connection with the same RDB name in the same activation group. However, it is possible to have multiple connections with different RDB names in the same activation group and to have multiple connections with the same Oracle server through different activation groups.

The Oracle Access Manager program is created using `ACTGRP (*CALLER)`. It, therefore, runs in the same activation group as the caller when it is invoked. When called from `STRSQL` or by the `CRTSQLxxx` commands, Oracle Access Manager may run in the default activation group. For more information about connection management and activation groups, refer to the IBM guide for DB2/400 SQL programming.

Release Information

This chapter contains release information for Oracle Access Manager for AS/400. It includes the following sections:

- [Product Set](#) on page 2-1
- [Changes and Enhancements](#) on page 2-1
- [Known Problems and Restrictions](#) on page 2-2

2.1 Product Set

The following products and component versions are included on the product CD-ROM. All components are at production level.

- Oracle Access Manager for AS/400: 10.2.0.1.0
- Oracle Net, TCP/IP adapter: 10.2.0.1.0

2.2 Changes and Enhancements

The following changes and enhancements have been made in release 10.2.0.1.0 of Oracle Access Manager for AS/400.

2.2.1 Release 10.2.0.1.0

The following changes have been made in this release of the Oracle Access Manager:

2.2.1.1 CURSOR WITH HOLD

DECLARE CURSOR statements can now include the WITH HOLD phrase.

2.2.2 Bugs Fixed for Release 10.2.0.1.0

The following bugs have been fixed for this release of the Oracle Access Manager for AS/400:

- 4969332
Intermittent connection problems using Access Manager; ORA-12154 failures
- 4907795
Function called once but runs twice and gets errors ORA-00001 and ORA-06512
- 4873885

- ORA-00020 Maximum number of processes exceeded
- 4771629
ORA00110, CPF7CFF errors when running CHGSQLNET under some CCSID
- 4683381
MCH3601 in AM400_RIDI at Stmt 61
- 4631655
AM400_RID Fails to run after changing system name
- 4610071
SQLSTATE 08000, SQLCODE -30080 when trying to CONNECT
- 4555789
AM400_RIDI job always has ORASRV0001 added to library list but other AM/400 programs use different ice program library
- 4006699
COBOL program fails with MCH3402 due to commitment control resource not being released at activation group end
- 3921461
Target Oracle Server runs out of processes
- 2966985
SQL0969 error when no precision is specified
- 5080676
Number format in Oracle, when access via AM is returning exponential format
NUMBER fields that Oracle describes as having a precision of 0 and a scale of -127 are now taken as having a precision and scale of 0.

2.3 Known Problems and Restrictions

The following problems and restrictions are known to exist in the Oracle Access Manager for AS/400 products on the product CD-ROM. The description of problems includes suggestions for dealing with them, when possible.

A current list of problems is available online. Contact your local Oracle Corporation office for information about gaining access to this list.

2.3.1 Known Problems

The following problems are known to exist in the Oracle Access Manager for AS/400.

2.3.1.1 STRSQL on Columns of Type NUMBER

When using STRSQL, an ORA-1457 might occur on columns of type NUMBER where no precision or scale is specified. Access Manager supplies default data types *precision* and *scale*, for these situations. Even the use of these data types, however, may not be sufficient, and you could still receive the ORA-1457 error. To check the default data type precision and scale values, use the Oracle Access Manager for AS/400 CHKSQLDFLT command.

The AS/400 is shipped with the following default settings:

```
Datatype *PKDEC (packed decimal)
Precision 7
Scale 2
```

For example, when using STRSQL with these settings, a column of type NUMBER with a value of 123456 results in an ORA-1457 error. This error occurs because the setting of 7 for precision with the setting of 2 for scale allows for only five digits to the left of the decimal point (precision - scale = 7 - 2 = 5). To avoid this error, you can change the default data type to *FLOAT or keep the data type as *PKDEC and change the precision to 8 while keeping the default scale of 2. This will allow six digits to the left of the decimal point (precision - scale = 8 - 2 = 6).

2.3.1.2 General Limitations

Oracle hints cannot be specified in a SQL statement that is explicitly part of any EXEC SQL statement. The IBM preprocessor strips out the Oracle hints, as if they were comments.

If you want Oracle hints in a DELETE, SELECT, or UPDATE statement, then you must:

1. Copy the text of the relevant DELETE, SELECT, or UPDATE statement to a host variable.
2. PREPARE a statement from that host variable.
3. For a SELECT, you must DECLARE a cursor for that prepared statement. Open the cursor, and then FETCH from that cursor.
4. For an UPDATE or DELETE, you must EXECUTE the prepared statement.

2.3.1.3 Stored Procedure Limitations

The following limitations are known to exist with Oracle stored procedures:

- The TIMESTAMP and TIME data types return the date only.
- You must use the DB2/400 DECLARE PROCEDURE statement in preparation to call an Oracle stored procedure from an AS/400 program.
- The DECLARE PROCEDURE statement must have the same number of arguments as the Oracle PL/SQL procedure. The input and output modes must also be the same.
- The SIMPLE CALL WITH NULLS phrase is not supported. The SIMPLE CALL phrase is supported.

2.3.2 Known Restrictions

The following restrictions are known to exist in the Oracle Access Manager for AS/400.

2.3.2.1 PREPARE with a COMMIT or ROLLBACK Command

When using two-phase commit processing, COMMIT or ROLLBACK commands PREPARED as dynamic SQL statements are rejected with the following error:

```
SQL0969: Error occurred while passing request to application requester driver program.
```

The following information is available in the job log:

```
ARDARPS: cannot PREPARE a COMMIT or ROLLBACK
```

For additional information on two-phase commit processing, refer to [Chapter 8, "Using Access Manager"](#).

2.3.2.2 RDBCNMTH Parameter set to *DUW

Access Manager requires that the RDBCNMTH parameter be set to *DUW for any SQL package that is preprocessed with any of the CRTSQLxxx commands.

2.3.2.3 Non-DML SQL Statements

When using two-phase commit processing, CONNECT is the only non-Data Manipulation Language (DML) SQL statement that is supported by Access Manager. Use of such non-DML statements results in the following error:

```
ORA-2089 COMMIT is not allowed in a subordinate session
```

2.3.2.4 COMMIT HOLD

The HOLD parameter on an EXEC SQL COMMIT HOLD command is not honored. All cursors are closed at each logical unit of work boundary (COMMIT and ROLLBACK).

2.3.2.5 LONG and LONG RAW Data Types

Only the first 32,740 bytes in a LONG or LONG RAW column can be retrieved. This is because of the DB2/400 length limit on a LONG VARCHAR column. Also note that the total data length on a SELECTed row is 32,760 bytes. The maximum length for a LONG or LONG RAW column will be further diminished by the amount of data in the OTHER columns that are retrieved in that row.

2.3.2.6 Installation into an IASP

The Access Manager can *not* be installed into a library that is part of an independent auxiliary storage pool (IASP). This is an IBM restriction on the placement of exit programs and subsystem descriptions. To get around this, just install the Access Manager in normal libraries on the machines/partitions that will be participating in the IASP. You should modify the user profiles of those users that will be using the Access Manager to retrieve DB2/400 data in an IASP so that the initial program (as defined in the User Profile) sets the current IASP group using the SETASPGRP command.

System Requirements

This chapter describes the hardware and software requirements of the Oracle Access Manager for AS/400. It includes the following sections:

- [Hardware Requirements](#) on page 3-1
- [Software Requirements](#) on page 3-2
- [Recommended Documentation](#) on page 3-2

3.1 Hardware Requirements

Oracle Access Manager for AS/400 requires the following hardware:

3.1.1 Processor

The Oracle Access Manager for AS/400 requires an AS/400 processor that supports any current version of OS/400. Also refer to Section 3.2.1, "[Operating System](#)", under Software Requirements.

3.1.2 CD-ROM Drive

The Oracle Access Manager for AS/400 is distributed on a CD-ROM.

- [Operating System](#)
- [Communications](#)
- [Oracle Server](#)
- [PTF Requirements](#)
- [Additional Software](#)

3.1.3 Disk Space

You should have at least 400 MB of free disk space available before beginning the installation of this product. At the end of the installation there will be approximately 52 MB of objects in the install library. There will be another 149 MB of objects in an Oracle Service Program Library which will have a name similar to `ORASRVnnnn` where the `nnnn` are numeric digits. Multiple installations of the Access Manager at the same level share a common Oracle Service Program Library. An installation of the Oracle Database Gateway at the same version level as the Access Manager will also share the same Oracle Service Program Library.

3.2 Software Requirements

Oracle Access Manager for AS/400 requires the following software:

3.2.1 Operating System

The Oracle Access Manager for AS/400 runs in OS/400 V5R2 or V5R3.

3.2.2 Communications

The Oracle Access Manager for AS/400 requires a TCP/IP connection to the target Oracle server.

3.2.3 Oracle Server

Any supported release of the Oracle server is supported by the Oracle Access Manager for AS/400. To be supported, the Oracle server must be configured with TCP/IP access.

3.2.4 PTF Requirements

For OS/400 V5R2, PTFs SI09769 and SI09772 should be applied. Oracle recommends that you keep up to date with the latest DB2/400 PTF group. You can check your current PTF status using the `WRKGRPPTF` command. For OS/400 V5R2 use `WRKGRPPTF SF99502` and for OS/400 V5R3, use `WRKGRPPTF SF99503`.

3.2.5 Additional Software

You must have the SQL Development Kit (IBM Program Number 5722-ST1) installed in order to run STRSQL and to develop embedded SQL programs.

3.3 Recommended Documentation

The following documents might be useful in providing additional information about products that are used with the Oracle Access Manager for AS/400:

- IBM Guide for DB2/400 SQL Programming
- *Oracle Call Interface Programmer's Guide*
- *Oracle C++ Call Interface Programmer's Guide*
- *Oracle Database Net Services Administrator's Guide*
- *Oracle Database Net Services Reference*
- *Oracle Database SQL Language Quick Reference*
- *Oracle Database SQL Language Reference*
- *SQL*Plus Quick Reference*
- *SQL*Plus User's Guide and Reference*

Installing Access Manager

The installation of the Oracle Access Manager for AS/400 is divided into two parts, installation and postinstallation. Before you begin installation, ensure that you have the required hardware and software that are described in [Chapter 3, "System Requirements"](#).

This chapter contains the following topics:

- [Checklists](#) on page 4-1
- [Installation](#) on page 4-2
- [Post-Installation](#) on page 4-4

4.1 Checklists

Use these checklists when installing the Oracle Access Manager for AS/400.

4.1.1 Installation Checklist

- [Step 1: Mount the Product CD-ROM](#)
- [Step 2: Start the Installation Process](#)
- [Step 3: Select the Installation Library Name](#)
- [Step 4: Verify Installation Parameters](#)
- [Step 5: Finish the Installation](#)

4.1.2 Post-Installation Checklist

- [Step 1: Add Oracle Access Installation Library to Library List](#)
- [Step 2: Change Password for User Profile](#)
- [Step 3: Configure Oracle Net](#)
- [Step 4: Test Connectivity to Oracle Server](#)
- [Step 5: Configure Access Manager](#)
- [Step 6: Configure Two-Phase Commit \(Optional\)](#)
- [Step 7: Retest Connectivity to Oracle Server \(Optional\)](#)

4.2 Installation

Before installing Oracle Access Manager for AS/400, you must log on with a user profile that has the *SECADM and *ALLOBJ special authorities. The user profile QSECOFR usually has that authority.

Caution: Oracle recommends that you back up any existing Access Manager library before upgrading to a new release.

The panels are shown as they appear when first displayed. The values that are documented in this installation guide are the default values that are provided by Oracle Corporation at the time that you install the Oracle Access Manager for AS/400. The default values are used unless they are manually changed before continuing.

4.2.1 Steps for Installation

- [Step 1: Mount the Product CD-ROM](#)
- [Step 2: Start the Installation Process](#)
- [Step 3: Select the Installation Library Name](#)
- [Step 4: Verify Installation Parameters](#)
- [Step 5: Finish the Installation](#)

4.2.2 Step 1: Mount the Product CD-ROM

Insert the product CD-ROM into the AS/400 CD drive.

4.2.3 Step 2: Start the Installation Process

Enter the following command:

```
LODRUN CD_drive
```

where *CD_drive* is the name of the CD drive where you mounted the product CD-ROM. The default is usually OPT01.

4.2.4 Step 3: Select the Installation Library Name

When the Install Access Manager Product Panel is displayed, use the default name of ORAAM400 or enter a name with a length from 1 to 10 characters long.

Example 4-1 Install Access Manager Product Panel

```
Install Oracle Access Manager product 10.2.0.1.0
```

```
Type Choices, press Enter.                               System: HQAS400
Install library name ..... ORAAM400                     Name (up to 10 characters)
Auxiliary storage pool id .... 1                          1-16
```

```
====>
```

```
Bottom
```

```
F1=Help    F4=Prompt    F9=Retrieve    F12=Cancel
```

(C) Copyright Oracle Corporation, 2006

The specified library name must denote an existing (but empty) library, or must denote a nonexistent library. The installation procedure will not install into a library with existing objects. The specified library name is also used to create an AS/400 User Profile. The password for that profile is also identical to the User Profile name. The password should be changed after installation using the CHGUSRPRF command.

For auxiliary storage pool ID, use the default of 1, or enter another ID.

After entering your information on the panel, press Enter to continue.

4.2.5 Step 4: Verify Installation Parameters

The following panel appears:

Example 4-2 Verify Access Manager Install Panel

```

Verify Oracle Access Manager Install at V10.2.0.1.0

                                           System:  HQAS400

Verify that you want to install the product.  Once this process has started, it
will create a library and other objects.  Enter *YES to verify this
operation before it is started.

New Install Library name .....      ORAAM400
Enter *YES to verify .....          *NO          *NO, *YES

===>                                           Bottom

F1=Help   F4=Prompt   F9=Retrieve   F12=Cancel
(C) Copyright Oracle Corporation, 2006

```

Enter *YES to continue the installation process. Press Enter to continue.

4.2.6 Step 5: Finish the Installation

If you entered *YES, then status messages will be displayed on the bottom of the screen as the installation proceeds. After a few minutes, a screen will appear, stating:

```
Install complete. Press ENTER. Check Job log for any errors.
```

When the installation is finished, use the following command to check the job log for error messages:

```
DSPJOBLOG
```

Installation of Oracle Access Manager for AS/400 is complete.

Note: During the installation, an AS/400 User Profile was created with the same name and password as the library in which the Access Manager was installed. You need to use the CHGUSRPRF command to change the password for that User Profile.

You need to change the password by using the following command syntax:

```
CHGUSRPRF USRPRF(install_lib) PASSWORD(new_password)
```

where:

install_lib is the installation library. The default is ORAAM400.

new_password is the new userid password that you want to use.

4.3 Post-Installation

This section describes the steps to be performed after the initial installation of Oracle Access Manager for AS/400.

4.3.1 Steps for Post-Installation

- [Step 1: Add Oracle Access Installation Library to Library List](#)
- [Step 2: Change Password for User Profile](#)
- [Step 3: Configure Oracle Net](#)
- [Step 4: Test Connectivity to Oracle Server](#)
- [Step 5: Configure Access Manager](#)
- [Step 6: Configure Two-Phase Commit \(Optional\)](#)
- [Step 7: Retest Connectivity to Oracle Server \(Optional\)](#)

4.3.2 Step 1: Add Oracle Access Installation Library to Library List

Before attempting to use any of the Access Manager configuration utilities, you must add the Access Manager library to your library list. To do this, enter the following command on any command line:

```
ADDLIBLE library_name
```

where *library_name* is the name of your Access Manager installation library.

The default position is *FIRST.

You do not need to manipulate the library list when running the Access Manager itself. Access Manager will determine which libraries need to be added to the library list, and it will add those libraries to the top of the user portion of the library list.

4.3.3 Step 2: Change Password for User Profile

Use the CHGUSRPRF command to change the password for the user profile that was created during the install. The name for that user profile is identical to the name of the library in which the Access Manager objects were installed. So, if you used the default instance name of ORAAM400, then Access Manager objects were placed into library ORAAM400, a user profile with a name of ORAAM400 was created with password ORAAM400.

4.3.4 Step 3: Configure Oracle Net

Configure Oracle Net on the Oracle integrating server. Refer to [Chapter 5, "Oracle Net"](#) for more information.

4.3.5 Step 4: Test Connectivity to Oracle Server

Refer to "[Connecting through the STRSQL \(Interactive SQL\) Commands](#)" on page 8-1 in [Chapter 8, "Using Access Manager"](#) for information about connecting to an Oracle server.

Note: You should test the connectivity before changing any configuration parameters.

4.3.6 Step 5: Configure Access Manager

Configure Oracle Access Manager for AS/400. All options have defaults that are used unless you change them. Refer to [Chapter 6, "Configuring Access Manager"](#) for more information.

Note: Steps 6 and 7 need to be performed only if you will be using two-phase commit.

4.3.7 Step 6: Configure Two-Phase Commit (Optional)

If you want to use two-phase commit processing, then you need to configure it now. Refer to [Chapter 7, "Distributed Transaction Coordination"](#) for the steps to use in configuring two-phase commit processing.

4.3.8 Step 7: Retest Connectivity to Oracle Server (Optional)

At this point, you can retest your connectivity to the Oracle server by using two-phase commit processing. Refer to "[Connecting through the STRSQL \(Interactive SQL\) Commands](#)" on page 8-1 for information about connecting to an Oracle server.

In the client-server model, a client application accesses a database server to satisfy database operations that are initiated by an application. The Oracle Access Manager for AS/400 is based upon the client-server model. All application processing, such as the user interface and data presentation, occurs at the client workstation on an AS/400. All Oracle database processing occurs at the Oracle server, which (when using Oracle Access Manager for AS/400) is *not* on an AS/400 computer.

Interaction between the client and server is achieved through a common dialogue or messaging system that is designed for the SQL language. With Oracle Net, databases and their applications can reside on different computers and communicate as peer applications in a Transparent Networking Substrate (TNS) network.

Read this chapter to learn how to configure and use Oracle Net with Oracle Access Manager for AS/400. It includes the following sections:

- [Oracle Net File Name Structure](#) on page 5-1
- [Connecting to Oracle through Oracle Net](#) on page 5-2
- [Defining the Oracle Server to Oracle Access Manager](#) on page 5-2
- [Oracle Net for TCP/IP AS/400 Configuration](#) on page 5-4

5.1 Oracle Net File Name Structure

Oracle Database Net Services Administrator's Guide and *Oracle Database Net Services Reference* refer to configuration, trace, and log files:

`basename.extension`

where:

`basename` is the base portion of the name

`extension` is the extension portion of the name.

An example of this form is `tnsnames.ora`.

On the AS/400, the Oracle Net parameter files are installed in the Access Manager library by using the following file name mapping:

- The base portion of the file name is mapped to a member in an AS/400 file.
- The extension portion of the file name is mapped to an AS/400 file.

For example, the `tnsnames.ora` parameter file is mapped to member `TNSNAMES` in the `ORA` file that is located in the Access Manager library.

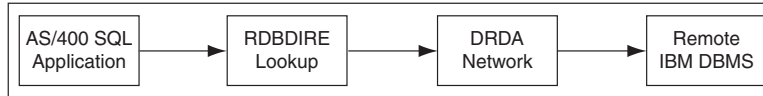
Member names are referred to as:

```
file(member_name)
```

5.2 Connecting to Oracle through Oracle Net

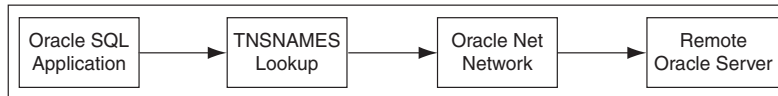
When an AS/400 application communicates with a remote IBM database server, it uses the IBM DRDA connection that is defined through the AS/400 `ADDRDBDIRE` command. `RDBDIRE` means Remote DataBase Directory Entry.

Figure 5–1 Native IBM Networking



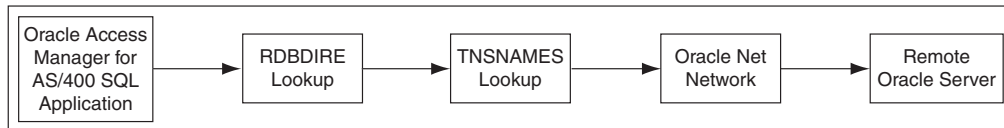
When an Oracle SQL application communicates with a remote Oracle server, it uses an Oracle Net connection that is defined in the `tnsnames.ora` file.

Figure 5–2 Native Oracle Networking



Connecting through Access Manager involves combining the IBM DRDA connection with the Oracle Net connection.

Figure 5–3 Oracle Access Manager Networking

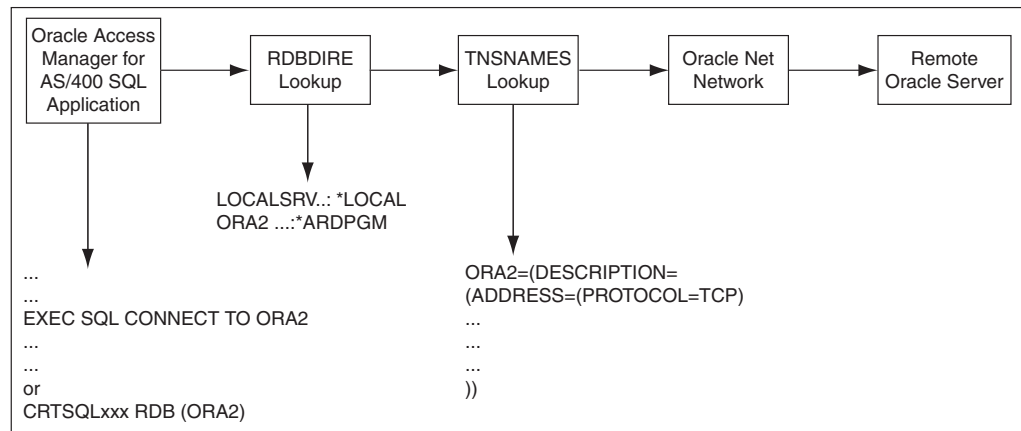


In [Figure 5–3](#), the remote database name is used as the key for both the `RDBDIRE` lookup and the `TNSNAMES` lookup.

5.3 Defining the Oracle Server to Oracle Access Manager

To connect to an Oracle server through Oracle Net, you must define an Oracle Net `TNSNAMES` entry with a name that matches your AS/400 `RDBDIRE` entry. This section provides a general description of these definitions. To create these definitions, refer to "[Oracle Net for TCP/IP AS/400 Configuration](#)" on page 5-4 for specific instructions. [Figure 5–4](#) shows how these matching entries interact:

Figure 5-4 Defining the Oracle Server to Access Manager



5.3.1 Remote Database Directory Entry

The AS/400 ADDRDBDIRE command is used to define remote database directory entries. The name you use on the ADDRDBDIRE command must be identical with the name of a description entry in your ORA (TNSNAMES) file. And that name is the same as the name used as a connection identifier in your CONNECT command either in your program or in STRSQL.

5.3.2 ORA(TNSNAMES)

To connect to the target Oracle server through Oracle Net, you must add an entry to the ORA (TNSNAMES) file in the Access Manager installation library. The entry name must match the name used in the RDB parameter when using the ADDRDBDIRE command. That is, the name must match the name of the RDB entry in the AS/400 RDB directory which is maintained by the ADDRDBDIRE command.

5.3.3 Configuration Entries

The following examples show the relationship between the AS/400 TCP/IP configuration and Access Manager. The ORA2 entries (in bold font) are the matching entry names.

- ORA (TNSNAMES)


```

ORA2 = (DESCRIPTION=
          (ADDRESS=
            (PROTOCOL=TCP)
            (PORT=1521)
            (HOST=UNIX9)
          )
          (CONNECT_DATA=(SERVICE_NAME=Oracle92)
          )
        )
      
```
- AS/400 Relational Database Directory Entry (DSPRDBDIRE)


```

Relational database . . . . . : ORA2
Remote location:
  Remote location . . . . . : *ARDPGM
Application requester driver:
  Program . . . . . : ORAARD
  Library . . . . . : ORAAM400
      
```

```
Interface level . . . . . : 1
Text. . . . . :
```

The TNSNAMES entry may be in any case, but it will be mapped to uppercase by Access Manager.

5.4 Oracle Net for TCP/IP AS/400 Configuration

Perform the following steps to configure the client connection between the AS/400 and the Oracle server. These configuration steps are presented here with the assumption that TCP/IP is already installed and configured on the AS/400 with basic connectivity (ping, for example) established between the AS/400 and the server that is hosting the Oracle Database server.

5.4.1 Steps for TCP/IP AS/400 Configuration

- [Step 1: Create Relational Database Directory Entry](#)
- [Step 2: Define TNSNAMES Entry](#)

5.4.2 Step 1: Create Relational Database Directory Entry

Using the AS/400 ADDRDBDIRE command, define a relational database directory entry (Panel 1, [Example 5-1](#)). Enter the name of the entry on the line titled `Relational database`. Enter `*ARDPGM` on the line titled "Name or address". Do not change the data on the line titled `Type`. You may change the `Text` parameter to whatever you choose. Press Enter. The screen will change to add a section entitled `Application Requester Driver` (Panel 2, [Example 5-2](#)).

Example 5-1 ADDRDBDIRE Panel 1

```

                                Add RDB Directory Entry (ADDRDBDIRE)

Type Choices, press Enter.

Relational database ..... ORACLETCP_____
Remote location:
Name or address ..... *ARDPGM_____
_____
_____
Type ..... *SNA          *SNA, *IP
Text ..... 'TCP/IP connection to oracle'_____
_____

                                Bottom

F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

In Panel 2 ([Example 5-2](#)), enter `ORAARD` on the line entitled "Program", and enter the name of the install library for the Access Manager on the line entitled "Library".

Example 5-2 ADDRDBDIRE Panel 2

```

                                Add RDB Directory Entry (ADDRDBDIRE)

Type Choices, press Enter.

Relational database ..... > ORACLETCP_____
Remote location:
  Name or address ..... > *ARDPGM_____
_____

Type ..... *SNA, *IP
Text ..... > 'TCP/IP connection to oracle'_____
_____

Application requester driver:
  Program ..... ORAARD_____ Name, *DRDA
  Library ..... ORAAM400___ Name, *LIBL, *CURLIB

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys

```

The following parameters must be entered:

`Relational database` is the name that you are assigning to the Oracle server relative to the AS/400. This must match the entry name of the ORA (TNSNAMES) entry.

`Remote location` must be *ARDPGM for the ARD interface.

`Program` must be ORAARD, the name of the Access Manager executable program.

`Library` must be the name of the Access Manager installation library.

You can also change the `text` parameter to whatever you choose.

5.4.3 Step 2: Define TNSNAMES Entry

Edit member TNSNAMES in the ORA file in the Access Manager installation library using the following command:

```
WRKMBRPDM library_name/ORA
```

where `library_name` is the name of your Access Manager installation library.

Enter "2" on the line for TNSNAMES to edit the TNSNAMES member. If you do not have the Program Development Manager, then you can use the EDTF command to edit the member.

The TNSNAMES entry name must match the name of the AS/400 RDB directory entry that was defined in the previous step using the ADDRDBDIRE command.

```

server_name = (DESCRIPTION=
                (ADDRESS=
                  (PROTOCOL=TCP)
                  (PORT=port_number)
                  (HOST=host_name)
                )
                (CONNECT_DATA=(SERVICE_NAME=service_name))

```

```
)  
)
```

where:

server_name is the name of a connection to an Oracle server that matches an entry in the RDB directory. An entry in the RDB directory can be added using the ADDRDBDIRE command.

TCP is the TCP protocol used for TCP/IP connections.

port_number is the port number of the Oracle Net listener. This is usually port number 1521.

host_name is the name that defines the system where the target Oracle server resides. This name must be in the local host definition on the AS/400 or in a name server on your network. Use option 10 of the CFGTCP command to edit the host table entries for your system. The host name can also be entered as an IP address, for example, 161.14.10.12.

service_name is the service name of the Oracle server.

Compare the instructions just provided with the following example:

```
ORA2 = (DESCRIPTION=  
      (ADDRESS=  
        (PROTOCOL=TCP)  
        (PORT=1521)  
        (HOST=UNIX9)  
      )  
      (CONNECT_DATA= (SERVICE_NAME=ORA2)  
    )  
  )
```

Configuring Access Manager

After installing Oracle Access Manager, you can:

- create a SQL package
- use the `STRSQL` command to interactively access data in an Oracle server

All Oracle Access Manager parameters or system values that require modification can be changed using Access Manager commands or through a menu system. These commands and their menus are described in this chapter. It includes the following sections:

- [Access Manager Commands](#) on page 6-1
- [CHGSQLDFLT](#) on page 6-2
- [CHGSQLNET](#) on page 6-5
- [CHGSQLDBG](#) on page 6-6
- [CRTORAPKG](#) on page 6-7
- [CHG2PCPRM](#) on page 6-9

6.1 Access Manager Commands

Before attempting to enter any of the Access Manager commands, you must add the Access Manager library to your library list. To do this, enter the following `ADDLIBLE` command on any command line:

Note: If you run the Oracle Access Manager, then it will automatically perform the `ADDLIBLE` and leave the required library in the library list. If you have not run the Oracle Access Manager, then Oracle Access Manager Commands (listed as the topics included in this chapter) will not run successfully until the library is added to the library list.

```
ADDLIBLE library_name
```

where *library_name* is the name of your Access Manager installation library. The default position is `*FIRST`.

Then, enter:

```
GO CMDORAAM
```

The command displays the Access Manager commands menu.

Example 6–1 Access Manager Commands Menu

```

CMDORAAM                Oracle Access Manager Commands                System: HQAS400

Select one of the following:

    1. Change Oracle Access Manager Default settings
    2. Change Oracle Access Manager Oracle Database settings
    3. Change Oracle Access Manager Debug settings
    4. Create an Oracle Access Manager SQL Package
    5. Change Oracle Access Manager Two-phase commit parameters

Selection or command

==>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
(C) Copyright Oracle Corporation, 2006

```

Each of the options on the Access Manager commands menu can be accessed directly by entering the command name from any command line. The commands are listed in the following table:

Table 6–1 Access Manager Commands

Menu Choice	Command Name	Purpose
1	CHGSQLDFLT	Changes the Access Manager default settings.
2	CHGSQLNET	Changes the Oracle Net trace settings.
3	CHGSQLDBG	Changes Access Manager debugging parameters.
4	CRTORAPKG	Creates a SQL package using the specified source and Oracle server name.
5	CHG2PCPRM	Changes Access Manager two-phase commit parameters.

6.2 CHGSQLDFLT

After selecting option 1 on the Access Manager commands menu (or after issuing the CHGSQLDFLT command), the first panel that is displayed confirms which library is to be changed. This should match the library where you installed Oracle Access Manager for AS/400.

Example 6–2 Access Manager Default Options Panel

```

                Oracle Access Manager Default Options                System: HQAS400

```

Type choices for default option, press Enter.

Existing Install library nameORAAM400 Name

====>

F3=Exit F4=Prompt F9=Retrieve F12=Cancel
(C) Copyright Oracle Corporation, 2006

The next panel displays the default user ID and default password that are needed for the Oracle server to which you are connecting. The defaults supplied on the distribution tape are SCOTT/TIGER. You will probably want to change this default for security reasons.

Example 6-3 Access Manager Default Options Panel

Oracle Access Manager Default Options

System: HQAS400

Type choices for default option, press Enter.

Existing Install library name	ORAAM400	Name
Access Manager default user id	SCOTT	Userid
Access Manager default password ...	*****	Password
Access Manager default blocksize ..	100	Blocksize
Access Manager default number format	*PKDEC	*FLOAT, *PKDEC
Access Manager default precision ..	9	0 - 31
Access Manager default scale	2	0 - Default precision
Access Manager NLS Language	AMERICAN_AMERICA.WE8EBCDIC37	
Access Manager Two-Phase Commit ...	*NO	*YES, *NO
Commitment-Control Resource Disposition	*RELEASE	*KEEP, *RELEASE

====> Bottom

F1=Help F4=Prompt F9=Retrieve F12=Cancel
(C) Copyright Oracle Corporation, 2006

The ten fields on the panel in [Example 6-3](#) are described as follows:

Existing Install library name identifies the library that contains the Access Manager program. The default is the Access Manager installation library.

Access Manager default user id specifies the default user ID. The default user ID and default password are used if the USER parameter is not specified in a CRTSQLxxx command or if the CONNECT TO statement is issued without the USER or USING options. If you are concerned about security, then you can change this field to BLANK. This forces you to always explicitly enter a user ID and password to CONNECT. If you

do set the user ID to blank and attempt to CONNECT without specifying a user ID, then you will receive an ORA-1017 error.

Access Manager default password specifies the default password for the default user ID. The default user ID and default password are used if the USER or the PASSWORD parameter is not specified in a CRTSQLxxx command, or if the CONNECT TO statement is issued without the USER or USING options.

Oracle recommends that you use the default password only for testing, and that in your application, you embed the user ID and password. A password, if given, will be encoded.

Access Manager default blocksize specifies the default block size. The Access Manager uses the block size data area to determine the number of rows to retrieve in a single fetch. The default is 100. A SELECT statement must contain the FOR FETCH ONLY clause in order for block fetching to be performed.

Access Manager default number format specifies the DB2/400 data type to use when Oracle describes an Oracle numeric field as being of type NUMBER with a scale and precision of zero.

*FLOAT specifies floating point notation. *PKDEC specifies packed decimal notation with precision set to the default precision, and with scale set to the default scale (refer to "[Access Manager default scale](#)" in this list). Oracle functions, such as AVG, COUNT(*), MAX, and SUM, return NUMBER values with precision and scale set to zero. For example, using the default precision and scale values of 9 and 2, respectively, the following SQL statements return the values stated here:

```
SELECT AVG(DEPTNO) FROM DEPTNO;
25.00 if *PKDEC is specified
2.50E+001 if *FLOAT is specified
SELECT 1/3 FROM DUAL;
.33 if *PKDEC is specified
3.33E-001 if *FLOAT is specified
```

*PKDEC is the default.

Access Manager default precision specifies the default precision to be used if *PKDEC is specified for Access Manager default number format. The default is 9.

Access Manager default scale specifies the default scale to be used if *PKDEC is specified for Access Manager default number format. The default is 2.

Access Manager Globalization Support language specifies the Globalization Support language. Globalization Support enables Oracle applications to interact with users in their native language using their conventions for displaying data. The default Globalization Support language can be changed to any valid value. Refer to [Appendix B, "Globalization Support"](#) for a list of valid values.

Access Manager Two-Phase Commit specifies whether two-phase commit processing is to be enabled. Two-phase commit processing ensures transactional consistency in a heterogeneous environment. If set to *NO, then data might not be synchronized between the local and remote databases. If set to *NO, then DDL statements, such as CREATE and DROP, may be used. If set to *YES, then DDL statements cannot be used.

If your application does *not* change data on either the AS/400 or on the Oracle server, then you can leave this option as *NO. Also, if you use the Oracle server only to get data (not to change data), then you can leave this option as *NO.

Commitment-Control Resource Release specifies when the commitment control resource (allocated to perform a job) is released. Oracle Access Manager for AS/400 is required (by IBM Access Requester Driver specifications) to use an AS/400 commitment control resource.

If you intend to use Access Manager within the confines of a user-defined commitment control definition, then you need Access Manager to release its commitment control resource when a transactional unit of work ends at a commitment control boundary. You reach a commitment control boundary when a COMMIT or ROLLBACK operation is performed. When Access Manager starts a new unit of work, it then reacquires the required commitment control resource. For this scenario, set Commitment-Control Resource Release to *RELEASE.

If you are not planning to use commitment control definitions external to Oracle Access Manager for AS/400, then you can specify *KEEP for this parameter.

6.3 CHGSQLNET

After selecting option 2 on the Access Manager commands menu (or after issuing the CHGSQLNET command), the first panel that is displayed confirms which library is to be changed. This should match the library where you installed Oracle Access Manager for AS/400.

Example 6-4 Access Manager Network Parameters Panel

```

Change Oracle Access Manager Network Parameters
                                                    System: HQAS400

Type choices for default option, press Enter.

Existing Install library name ..... ORAAM400           Name

====>

F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel
(C) Copyright Oracle Corporation, 2006

```

On the next panel, you can change the settings of the Oracle Net trace options. Because running an Oracle Net trace can adversely affect AS/400 and Access Manager performance, do not change these settings unless you are asked to do so by Oracle Support Services.

Example 6-5 Access Manager Network Parameters Panel

```

Change Oracle Access Manager Network Parameters

```

System: HQAS400

Type choices, press Enter.

```
Existing Install library name ..... ORAAM400      Name
Client trace level ..... *OFF                    *OFF, *USER, *ADMIN
```

====>

```
F1=Help   F4=Prompt   F9=Retrieve   F12=Cancel
(C) Copyright Oracle Corporation, 2006
```

The fields on the panel in [Example 6-5](#) are described as follows:

Existing Install library name identifies the library that contains the Access Manager program. The default is the Access Manager installation library.

Client trace level specifies the tracing level for the AS/400 client. Valid values are *OFF, *USER, *ADMIN, and 16. The number 16 gives full trace information. Additional values can be supplied by Oracle Support Services for debugging purposes.

6.4 CHGSQLDBG

After selecting option 3 on the Access Manager commands menu (or after issuing the CHGSQLDBG command), the first panel that is displayed confirms which library is to be changed. This matches the library where you installed Oracle Access Manager for AS/400.

Example 6-6 Access Manager Debugging Options Panel

Change Oracle Access Manager Debugging Options

System: HQAS400

Type choices for debugging options, press Enter.

```
Existing Install library name ..... ORAAM400      Name
```

Bottom

====>

```
F1=Help   F4=Prompt   F9=Retrieve   F12=Cancel
(C) Copyright Oracle Corporation, 2006
```

On the next panel, you can change the debugging settings. The debugging function helps diagnose Access Manager problems. Do not change these settings unless you are asked to do so by Oracle Support Services.

Example 6–7 Access Manager Debugging Options Panel

```

Change Oracle Access Manager Debugging Options
System: HQAS400

Type choices for debugging options, press Enter.

Existing Install library name ..... ORAAM400           Name
Access Manager debugging control string .. 0           Debug String
Access Manager trace file ..... ARDTRACE              File
Access Manager trace library ..... ORAAM400           Library

Bottom

====>

F1=Help   F4=Prompt   F9=Retrieve   F12=Cancel
(C) Copyright Oracle Corporation, 2006

```

The fields on the panel in [Example 6–7](#) are described as follows:

Existing Install library name identifies the Access Manager installation library.

Access Manager debugging control string specifies the level of tracing. A 0 (zero) turns off all tracing and a 1 turns on full tracing. The trace output always goes to a member in the file that is determined by the Access Manager trace file and the Access Manager trace library.

Access Manager trace file specifies the qualified file name for trace output under control of the Access Manager debugging control string. The file needs to be created as a SOURCE PHYSICAL FILE with a maximum record length of 240.

Access Manager trace library specifies the library portion of a qualified file name. This field is used along with the Access Manager trace file field to designate a file into which trace members are written. If specified as blank, then a value of QTEMP is used.

This field is relevant only if the Access Manager debugging control string field contains a nonzero value.

6.5 CRTORAPKG

After selecting option 4 on the Access Manager commands menu (or after issuing the CRTORAPKG command), the Oracle SQL Package panel is displayed. Access Manager provides CRTORAPKG to facilitate the development of distributed applications that are targeted for execution against an Oracle server. Access Manager issues the necessary commands to precompile, compile, and bind your application program. Refer to [Chapter 8, "Using Access Manager"](#) for information about sample programs.

Example 6–8 Oracle SQL Package Panel

```

                                Create Oracle SQL Package
                                System: HQAS400

Type choices, press Enter.

File ..... QSQLCSRC           name
  Library ..... ARDQA           name
Member ..... XI1               name
Language Type ..... *C         *C, *RPG, *COBOL
Oracle server ..... ORACLETCP
Activation Group ..... *New
Object Type ..... *PGM
Batch/Terminal Compile ..... *BATCH
Target Release ..... *CURRENT
Additional pre-compile options .....

====>

F1=Help   F4=Prompt   F9=Retrieve   F12=Cancel
(C) Copyright Oracle Corporation, 2006

```

The required input fields on the panel in [Example 6–8](#) are as follows:

`File` is the name of a file containing the program source.

`Library` is the name of the library containing the file.

`Member` is the source member in the file.

`Language type` is the language type. Valid values are *C, *COBOL, and *RPG.

`Oracle server` is the name of the target Oracle server. This must match an entry name in `ORA(TNSNAMES)` and an entry in the RDB directory.

After filling in the data, the file, library, and member entries are verified to ensure that they exist.

Caution: During compilation, CRTORAPKG attempts to connect to the specified Oracle server. Ensure that your Oracle server and your network connection are active before running CRTORAPKG. You can use STRSQL to check the validity of your connection.

After the compile is complete, check the job log to determine if the package creation was completed successfully. Check the output spool file for the compile. Multiple spool files will exist for each compile. If package creation failed, then check for the following problems:

- syntax errors in the program source
- missing RDB directory entry
- missing or incorrect Oracle server entry name in `ORA(TNSNAMES)`

- unavailable target Oracle server
- nonfunctional Oracle Net system

The CRTORAPKG command uses the relevant IBM CRTSQL_{xxx} command to pre-process your file. Failure will occur only if the severity level of an error is greater than 30. Some severity level 30 messages are issued because the preprocessor does not recognize some Oracle SQL constructs or functions. For this reason, you must examine the spooled output to determine the validity of any errors of severity level 30.

The IBM precompiler flags any reference to Oracle server sequence numbers (error SQL5001, severity 30) and any reference to non-IBM functions, such as TO_CHAR and TO_DATE (error SQL0105, severity 30), as errors.

After finding and correcting the error, resubmit the job from the CRTORAPKG panel. Ensure that the input information is correct before pressing Enter.

6.6 CHG2PCPRM

This command allows you to change Access Manager two-phase commit parameters. The following panel is displayed:

Example 6–9 Change RID (Resolve In Doubt) Options Panel

```

                                Change RID (Resolve In Doubt) Options
                                System: HQAS400

Installation Library: ORAAM920

DataQueue Timeout (in seconds) ..... 90
CleanUp Delay (in seconds) ..... 30
Trace Level ..... 49
Dead Check Counter ..... 100

Type options, press Enter.
  1-Add  2=Change  4=Remove

Opt   Oracle Server Name   Description
-----
      VMS_SRV1             Connect to VMS1X system
      MVS03                MVS system 3
      MVS_92               MVS running Oracle 9.2
      PC_GTW_92            WinNT 4.0 server - 9.2
      H920_130             SUN running Oracle 9.2.0

                                Bottom

F1=Help  F4=Prompt  F9=Retrieve  F10=Switch to PrivID screen  F12=Cancel

(C) Copyright Oracle Corporation, 2006

```

For more information about this panel, refer to [Chapter 7, "Distributed Transaction Coordination"](#).

Distributed Transaction Coordination

Read this chapter to learn about two-phase commit processing. It includes the following sections:

- [Two-Phase Commit Overview](#) on page 7-1
- [Two-Phase Commit and Oracle Access Manager for AS/400](#) on page 7-2
- [Two-Phase Commit Process](#) on page 7-2
- [In-Doubt Transactions](#) on page 7-3
- [Enabling Two-Phase Commit Processing](#) on page 7-4
- [Oracle Required Privileges](#) on page 7-8
- [AM400_COM Program](#) on page 7-9

7.1 Two-Phase Commit Overview

Oracle Access Manager for AS/400 supports two-phase commit processing to ensure transaction consistency in a heterogeneous environment. Two-phase commit processing is an optional feature in Oracle Access Manager.

During installation, the Access Manager parameter for two-phase commit processing defaults to *NO. To enable transaction consistency in a heterogeneous environment, the two-phase commit parameter must be changed to *YES on the Change Access Manager Default Options panel or by using the CHGSQLDFLT command. For additional information, refer to [Chapter 6, "Configuring Access Manager"](#).

Note: If the two-phase commit parameter is set to *NO, then data might not be synchronized between local and remote databases. Synchronization is needed *only* when changes in data (changes in files or tables) are being made in a transaction both on the AS/400 and on the remote Oracle server.

7.1.1 Terminology

AM400_RID program automatically resolves in-doubt transactions. This program must be active to enable two-phase commit processing. The AM400_RID program runs in a subsystem which has the same name as the install library.

AM400_COM program provides a mechanism for privileged users to dynamically communicate with the AM400_RID program.

Two-phase commit parameter enables two-phase commit processing in Oracle Access Manager for AS/400. The default setting is *NO. It must be changed to *YES to enable two-phase commit processing.

Commit point site is the processing node that initiates a commit or roll back. The AS/400 that is participating in a two-phase commit transaction is always designated as the commit point site. The commit point site is also known as the "coordinator" site.

Prepare phase is the phase during which the coordinator asks participants to prepare to commit or to roll back the changes that were made during a transaction.

Commit phase is the phase during which the coordinator asks all nodes to commit the transactions, if all participants have responded to the coordinator that they are prepared to do so. If any participants cannot prepare, then the coordinator asks all nodes to roll back the transaction.

Implied commit When a program (activation group) ends without error, default behavior for the AS/400 system triggers all pending updates to be committed. This commitment of all pending updates would include updates to Oracle data as part of the AS/400 transactions through the Oracle Access Manager for AS/400.

7.2 Two-Phase Commit and Oracle Access Manager for AS/400

Oracle Access Manager for AS/400 enables AS/400 applications to update data on both the AS/400 and remote Oracle servers during a single transaction. Any time that updates are allowed in a distributed environment, a new level of complexity is introduced. When multiple users are sharing and accessing data that exists at many sites, rather than at a single site, these distributed transactions require careful management.

With Oracle Access Manager for AS/400, you now have the option to ensure that all participants (nodes) in a distributed transaction react unanimously to the action to take when a transaction ends. That is, they all commit, or they all roll back.

For a database to fully participate in a two-phase commit transaction, it must support both a prepare phase and a commit phase for communicating transactions. The prepare phase ensures that all participating nodes that are referenced in a distributed transaction are prepared to commit or to abort the transaction, regardless of intervening failures. The prepare phase is performed when a COMMIT command is issued at the end of a distributed transaction. The commit phase is performed when all transaction participants have responded to the coordinator that they are prepared to commit.

The Oracle server supports two-phase commit transactions. Therefore, any number of Oracle servers can participate in a distributed two-phase commit transaction.

Note: If the two-phase commit parameter has been set to *YES, and if the AM400_RID program is active, then the prepare phase occurs automatically when an application COMMIT is executed. No other action is necessary.

7.3 Two-Phase Commit Process

There are two phases for two-phase commit processing:

1. prepare phase

2. commit phase

7.3.1 Phase 1: Prepare

When DB2/400 receives a commit request from an application or from STRSQL, it asks all participants in the distributed system to enter a prepared state (to promise to commit or roll back the transaction, even if there is a failure). The participating sites reply either that they are prepared to commit or that they are not prepared to commit. When Access Manager receives such a commit request from DB2/400, it broadcasts an informational message to the AM400_RID program. This communication is one way. The AM400_RID program sends *no* communication to the Access Manager.

7.3.2 Phase 2: Commit

If the AS/400 receives a `prepared to commit` message from all the participating sites, then the AS/400 broadcasts a `COMMIT` command to all sites. If even one site has replied that it is not prepared to commit, then the AS/400 aborts the transaction and broadcasts a `ROLLBACK` command. When an Access Manager site has committed, an information message is sent to the AM400_RID program to announce the end of the transaction.

Note: Access Manager operates under AS/400 commitment control, including implied commits (defined in section "[Implied commit](#)" on page 7-2). Refer to the appropriate IBM AS/400 documentation for additional information.

7.4 In-Doubt Transactions

An in-doubt transaction occurs when a program or operation terminates abnormally before committing or rolling back the transaction. This creates an outstanding transaction, which must be resolved in order to free the involved resources and to ensure transaction consistency.

7.4.1 Resolving In-Doubt Transactions

In-doubt transactions can be resolved automatically by Oracle Access Manager for AS/400. Oracle Access Manager uses a separate task to resolve in-doubt transactions. This task runs in its own subsystem and must be active to enable two-phase commit processing. The task runs the AM400_RID program. On the `WRKACTJOB` display, the AM400_RIDI job will be seen to be active in the "lib" system (where "lib" is the name of the install library).

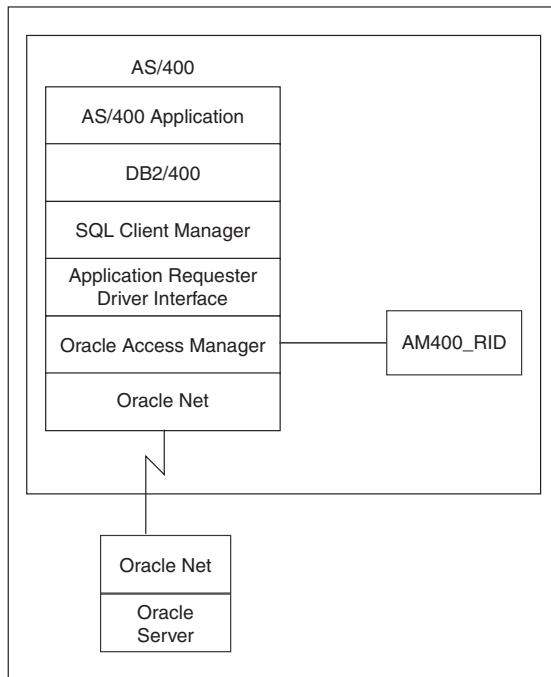
The AM400_RID program controls the automatic resolve-in-doubt processing for Oracle Access Manager for AS/400. This program runs in its own subsystem parallel to user programs. It is not called by user programs. The ability to perform resolve-in-doubt processing is required if two-phase commit was requested on the `CHGSQLDFLT` command through the "Access Manager Two-Phase Commit" line by specifying `*YES` to that option. The AM400_RID program performs resolve-in-doubt processing *only* for those Oracle Servers to which it has a valid connection. The only Oracle Servers to which the AM400_RID program might have a valid connection are those named in the list of Oracle Servers on the first page of the `CHG2PCPRM` command (the AM400_COM, see later, can actually be used to dynamically add Oracle Servers to that list). When a user attempts to `CONNECT`, the Access Manager queries the status of the AM400_RID program to see if it has a valid connection to the same Oracle Server

to which the user is attempting to CONNECT. If it does, then the user connection is allowed. If it does not have a valid connection, then the user CONNECT fails.

Note: Resolve-in-doubt processing actually occurs very rarely. It usually occurs due to some catastrophic failure. But it is required in order to keep data in a consistent state among the systems performing a distributed transaction. Informational messages are sent from the Access Manager program to the AM400_RID program every time a transaction changes status. No user data is sent in these messages.

The Oracle user ID specified for each connection that AM400_RID has with an Oracle Server must have rather high privileges. User ID (and password) are specified on the second screen of the CHG2PCPRM command when you are adding or modifying a possible connection. See Section 7.6, Oracle required privileges for further information.

Figure 7-1 AS/400 In-Doubt Transactions



As a transaction progresses, the Access Manager sends status messages to the AM400_RID program. The AM400_RID program does not send anything back to the Access Manager. In this manner, the AM400_RID program is kept informed of the status of all Oracle Access Manager transactions.

7.5 Enabling Two-Phase Commit Processing

To enable two phase commit processing, perform the following steps.

7.5.1 Checklist for Enabling Two-Phase Commit

- [Step 1: Change Access Manager Two-Phase Commit Default](#)

- [Step 2: Change the AM400_RID Options](#)
- [Step 3: Add or Remove AM400_COM Privileged User IDs](#)
- [Step 4: Start the Subsystem that Runs the AM400_RID Program](#)
- [Step 5: Start the AM400_COM Program \(Optional\)](#)

7.5.2 Step 1: Change Access Manager Two-Phase Commit Default

Change the Access Manager two-phase commit parameter to *YES. For more information about this panel and its settings, refer to [Chapter 6, "Configuring Access Manager"](#).

7.5.3 Step 2: Change the AM400_RID Options

After issuing the CHG2PCPRM command, the first panel that is displayed confirms which library is to be changed and confirms the four resolve-in-doubt (RID) single-valued options that can be manipulated. The library name should match the library where you installed the Oracle Access Manager for AS/400. These four single-valued options are listed in the panel as: "DataQueue Timeout (in seconds)", "CleanUp Delay (in seconds)", "Trace Level", and "Dead Check counter". After these single-valued options, space is available on the panel to describe the Oracle servers to which Oracle Access Manager for AS/400 will connect in response to a connection request in STRSQL or in user applications.

Example 7-1 Change Resolve In Doubt (RID) Options Panel

```

Change RID (Resolve In Doubt) Options
                                     System: HQAS400

Installation Library:  ORAAM130

DataQueue Timeout (in seconds)  . . . . . 90
CleanUp Delay (in seconds)      . . . . . 30
Trace Level . . . . . 49
Dead Check Counter . . . . . 100

Type options, press Enter.
  1=Add 2=Change 4=Remove

Opt   Oracle Server Name   Description
-----
VMS_SRV1   Connect to VMS1X system
MVS_920    MVS running Oracle 9.2.0
MVS_901    MVS running Oracle 9.0.1
PC_GTW_92  WinNt 4.0 server - 9.2
H920_130   SUN running Oracle 9.2.0

                                     Bottom

F1=Help  F4=Prompt  F9=Retrieve  F10=Switch to PrivID screen  F12=Cancel

(C) Copyright Oracle Corporation, 2006

```

Caution: The `CLEANUP_DELAY`, `DATAQUEUE_TIMEOUT`, and `DEAD_CHECK_COUNTER` are tuning parameters that are used to govern how often and when the `AM400_RID` program attempts to perform resolve in-doubt tasks. Do not change these parameters unless directed to do so by Oracle Support Services.

`Oracle Server Name` defines the name of any Oracle server that is to be accessed by Access Manager. If a server is not found in this list of servers, then it cannot be accessed by Access Manager when two-phase commit is enabled. When resolving an in-doubt transaction, the `AM400_RID` program uses this list to identify the Oracle server to which it must communicate. The Oracle server name is the same as the AS/400 RDB directory name and the TNS entry name.

The tuning parameters are described below:

`DATAQUEUE_TIMEOUT` is a tuning parameter that requires a single numeric value (in seconds). With two-phase commit enabled, every Oracle Access Manager user task transmits a status signal to the `AM400_RID` program for every connect and disconnect and at the beginning and end of every transaction. If a status signal is not received by the `AM400_RID` program after a specified interval, then a timeout occurs.

The `AM400_RID` program will check the status of all jobs (about which it knows) at least one time in every `n` seconds, where `n` is specified by the `DATAQUEUE_TIMEOUT` parameter. If a connection has already been terminated when the timeout occurs, then the `AM400_RID` program attempts to perform resolve in-doubt tasks.

`CLEANUP_DELAY` is a tuning parameter that requires a single numeric value (in seconds). It specifies the amount of time that the `AM400_RID` program waits (after it has determined that resolve-in-doubt processing might be needed) before attempting to perform resolve in-doubt tasks.

`TRACE_LEVEL` determines the trace level for the `AM400_RID` program. You should change this parameter only when requested to do so by Oracle Support Services.

`DEAD_CHECK_COUNTER` is a tuning parameter that requires a single numeric value. It specifies the minimum (or threshold) number of status signals that will be received by the `AM400_RID` program before it attempts to perform resolve in-doubt tasks. If the user connection still exists, then no action is taken.

Note: The `DATAQUEUE_TIMEOUT` and `DEAD_CHECK_COUNTER` together determine how often the `AM400_RID` program checks to see if an in-doubt transaction might be present. `DATAQUEUE_TIMEOUT` specifies a value in time space, and `DEAD_CHECK_COUNTER` specifies a value in event-count space.

To add a new Oracle server, select option 1. To change the attributes of an existing Oracle server, select option 2. After selecting either option, the following panel is displayed:

Example 7-2 Change or Enter New Server Attributes Panel

```
Change or Enter New Server Attributes                                     System: HQAS400
Installation Library . . . . . ORAAM130
Oracle Server Name . . . . . VMS_SRV1
```

```

Privileged Oracle Id . . . . . VMS1Z
Password for Oracle Id . . . . . zilchnut
Startup Parameter . . . . . startup
Description . . . . . Connect to VMS1X system
Dead Check Counter . . . . . 100

```

F1=Help F4=Prompt F9=Retrieve F12=Cancel

(C) Copyright Oracle Corporation, 2006

The fields in the panel are described as follows:

`Oracle Server Name` defines the name of the Oracle server that is to be accessed by Oracle Access Manager for AS/400.

`Privileged Oracle Id` designates the Oracle user ID that is associated with the defined server name. For additional information, refer to ["Oracle Required Privileges"](#) on page 7-8.

`Password for Oracle Id` is the password of the `Privileged Oracle Id` that is associated with the defined server name. This parameter will be saved in an encrypted form. For additional information, refer to ["Oracle Required Privileges"](#) on page 7-8.

`Startup Parameter` specifies when the `AM400_RID` program can attempt to connect to the designated server. If the `STARTUP` keyword is specified, then the `AM400_RID` program attempts to connect to the designated server when the `AM400_RID` program is started. If no keyword is specified, then the `AM400_RID` program attempts to connect to the designated server when any applications tries to connect to the server for the first time.

`Description` is a description of the defined `oracle_server`.

7.5.4 Step 3: Add or Remove AM400_COM Privileged User IDs

The options found on the panel in [Example 7-1, "Change Resolve In Doubt \(RID\) Options Panel"](#) can also be dynamically changed using the `AM400_COM` program. Use of the `AM400_COM` program to change these options is restricted to the set of AS/400 user IDs found in the panel in [Example 7-3](#).

The `AM400_COM` program allows specified AS/400 user profiles to dynamically add or remove Oracle server parameters, to designate `AM400_RID` connections to Oracle servers, and to shut down connections to those Oracle servers. For example, if an Oracle server is not designated in the "Oracle Server Name" column in the Change RID (Resolve In Doubt) Options panel, then it can be defined dynamically (for the current invocation of the `AM400_RID` program) with the `AM400_COM` program. Refer to ["AM400_COM Program"](#) on page 7-9 for additional information about the `AM400_COM` program.

To access the Change RID (Resolve In Doubt) Privileged IDs panel, first issue the `CHG2PCPRM` command. The first panel that is displayed confirms which library is to be changed and confirms the four single-valued RID options that can be used. Then select PF10 to display the Change RID (Resolve In Doubt) Privileged IDs panel.

Example 7-3 Change Privileged IDs (Relative to AM400_COM) Panel

```

Change Privileged IDs (Relative to AM400_COM)
System: HQAS400

Installation Library:  ORAAM130

Time options, press Enter.
  1=Add  4=Remove

Opt      Privileged ID
        QSECOFR
        AM400ADMIN

Bottom

F1=Help  F4=Prompt  F9=Retrieve  F10=Switch to OracleServers screen  F12=Cancel

(C) Copyright Oracle Corporation, 2006

```

The user ID in the `Privileged ID` field has full access to the `AM400_COM` program.

The `Change Privileged IDs (Relative to AM400_COM)` panel is distributed with the default of `QSECOFR` in the `Privileged ID` field. `QSECOFR` is the AS/400 security administration user ID.

7.5.5 Step 4: Start the Subsystem that Runs the AM400_RID Program

To enable two-phase commit processing, the subsystem that runs the `AM400_RID` program must be started using the following command:

```
STRSBS install_lib/install_lib
```

where `install_lib` is the library in which Access Manager is installed.

7.5.6 Step 5: Start the AM400_COM Program (Optional)

You can now start the `AM400_COM` program to dynamically change the current options for the `AM400_RID` program. The changes made by the `AM400_COM` program are not persistent. They are valid only for the current instance of the `AM400_RID` program. Refer to "[AM400_COM Program](#)" on page 7-9 for more information about starting the `AM400_COM` program.

7.6 Oracle Required Privileges

For the `AM400_RID` program to be able to perform resolve in-doubt processing, it must be able to connect to any Oracle server to which typical users connect with `STRSQL` or with a SQL-program package. The `AM400_RID` program connects to the Oracle server that is designated by `Oracle Server Name` in the panel in [Example 7-2](#) on page 7-6. The `AM400_RID` program connects to the Oracle server by using the Oracle user ID and password that are specified for that server on the `CHG2PCPRM` panel ([Example 7-2, "Change or Enter New Server Attributes Panel"](#)). The privileged user ID must have the following privileges:

- read access to the `SYS.PENDING_TRANS$` table

- update access to the `SYS.PENDING_SESSIONS$` table
- privileges to `COMMIT` or `ROLLBACK` any transaction with the `FORCE` option. For more information about the `FORCE` option, refer to the *Oracle Database SQL Language Reference*.

To grant these privileges, the data base administrator (DBA) can issue the following `GRANT` commands in Oracle by using `SQL*Plus`. This can be done for every server that is designated in the `Oracle Server Name` column in the Change RID (Resolve In Doubt) Options panel (Example 7-1).

```
GRANT CREATE SESSIONS, FORCE ANY TRANSACTION TO userid;
GRANT SELECT ON SYS.PENDING_TRANS$ TO userid;
GRANT UPDATE ON SYS.PENDING_SESSIONS$ TO userid;
```

where *userid* is the Oracle user ID for the associated server, as set or changed in the Change or Enter New Server Attributes panel.

7.7 AM400_COM Program

The `AM400_COM` program is used to communicate dynamically with the `AM400_RID` program. The `AM400_RID` program controls the automatic resolve in-doubt processing of Oracle Access Manager for AS/400. The `AM400_COM` program can query the `AM400_RID` program for the following information:

- the servers to which the `AM400_RID` program is connected
- the user tasks connected to the various Oracle servers

The `AM400_COM` program can also be used to dynamically alter the environment of the `AM400_RID` program during an AS/400 user session. The `AM400_COM` program can be used to perform the following tasks:

- add Oracle servers to the list of servers to which normal Access Manager tasks can connect
- remove Oracle servers from the list of available servers so that no more connections can be made from AS/400 user tasks to those servers
- change the `DATAQUEUE_TIMEOUT` parameter
- change the `CLEANUP_DELAY` parameter

To access the `AM400_COM` program, type the following on the AS/400 command line:

```
CALL AM400_COM PARM(install_lib)
```

where *install_lib* is the library in which Access Manager is installed.

7.7.1 AM400_COM Commands

The `AM400_COM` program recognizes the following commands:

`STATUS` requests global connection status of the `AM400_RID` program. Examples:

- Request the status of all servers to which the `AM400_RID` program has a connection:

```
STATUS SERVERS
```

or

STATUS

- Request the status of the connection between AM400_RID and Oracle server *server_name*:

```
STATUS SERVER server_name
```

- Request the status of what is known about a given job: The *numeric_value* is the six digit AS/400 job number.

```
STATUS JOB numeric_value
```

ADD adds a server to the list of available servers. For example:

```
ADD SERVER server_name oracle_id oracle_password
```

REMOVE removes a server from the list of available servers. For example:

```
REMOVE SERVER server_name
```

STARTUP starts a connection between the AM400_RID program and an Oracle server. For example:

```
STARTUP SERVER server_name
```

SHUTDOWN shuts down a connection between the AM400_RID program and an Oracle server. For example:

```
SHUTDOWN SERVER server_name [FORCE]
```

MODIFY modifies the Oracle user ID and the password to be used when the AM400_RID program attempts to connect to a specific Oracle server. For example:

```
MODIFY SERVER server_name oracle_id oracle_password
```

QUERY queries various items within the AM400_RID environment. Examples:

1. Query whether the AM400_RID program knows about a given server or has a connection to a given server. If the STARTUP parameter is given, and if the AM400_RID program knows about the server, but no connection exists from the AM400_RID program to that server, then an attempt is made to start the connection. For example:

```
QUERY SERVER server_name [STARTUP]
```

2. Query the value that is set for the DATAQUEUE_TIMEOUT parameter. For example:

```
QUERY DATAQUEUE_TIMEOUT
```

3. Query the value that is set for the CLEANUP_DELAY parameter. For example:

```
QUERY CLEANUP_DELAY
```

4. Query the value that is set for the DEAD_CHECK_COUNTER parameter. For example:

```
QUERY DEAD_CHECK_COUNTER
```

5. Query the value that is set for the TRACE_LEVEL parameter. For example:

```
QUERY TRACE_LEVEL
```

6. Query the value that is set for the SERVER_STARTUP_TIMEOUT parameter. This value is used by the AS/400 user task when it is attempting to ensure that the AM400_RID program has a connection to the same Oracle server to which the user task is attempting to connect. If the user task does not get a positive response from the AM400_RID program within the timeout value that is set by SERVER_STARTUP_TIMEOUT, then the user task is unable to CONNECT to the Oracle server. For example:

```
QUERY SERVER_STARTUP_TIMEOUT
```

SET sets various items within the AM400_RID environment. The following items can be set:

```
SET CLEANUP_DELAY numeric_value
SET DATAQ_TIMEOUT numeric_value
SET DEAD_CHECK_COUNTER numeric_value
SET TRACE_LEVEL numeric_value
SET SERVER_STARTUP_TIMEOUT numeric_value
```

LIST displays various items in the AM400_RID environment. Examples:

1. A list of all the servers that are known to the AM400_RID program and whether or not a connection exists from the AM400_RID program to that server. For example:

```
LIST AVAILABLE SERVERS
```

2. A list of various items about the shared data space through which the AM400_COM program, the AM400_RID program, and the AS/400 user tasks communicate. Use of this command might be requested by Oracle Support Services. An example:

```
LIST DATASPACE HEADER
```

3. A list of all used entries in the shared data space. Use of this command might be requested by Oracle Support Services. An example:

```
LIST ALL DATASPACE ENTRIES
```

FORCE forces a scan of all AS/400 user jobs that are known to the AM400_RID program and marks any jobs that need to use resolve in-doubt processing. For example:

```
FORCE SCAN
```

Using Access Manager

To use Oracle Access Manager for AS/400, you need to understand the following topics:

- [Connecting to the Oracle Server](#) on page 8-1
- [Connection Management](#) on page 8-3
- [Describing Tables](#) on page 8-4
- [Stored Procedures](#) on page 8-4
- [Conversion of Oracle Data Types to DB2/400 Data Types](#) on page 8-6
- [Message and Error Code Processing](#) on page 8-7

8.1 Connecting to the Oracle Server

When the network definition is completed, you can connect to an Oracle server from interactive SQL or from the precompiled AS/400 program.

8.1.1 Connecting through the STRSQL (Interactive SQL) Commands

From the AS/400 command line, enter:

```
STRSQL
```

Ensure that your interactive STRSQL session uses the SQL syntax `library.file`. To ensure this, use `STRSQL NAMING(*SQL)`. Access Manager does not support the AS/400 `library/file` syntax.

From the interactive SQL command line, enter the IBM DB2/400 `CONNECT TO` command in order to connect to the Oracle server:

```
CONNECT TO remote_oracle USER userid USING 'pw'
```

where:

`remote_oracle` is your RDB directory entry that has a matching `TNSNAMES` entry.

`userid` is a valid Oracle user ID.

`pw` is the password for the Oracle user ID.

If no user ID is specified, then Access Manager uses the default user ID and password. The default user ID and password are `SCOTT` and `TIGER`. You can change these with the `CHGSQDFLT` command. Refer to [Chapter 6, "Configuring Access Manager"](#) for more information.

After you are connected, you can issue SQL statements to the remote Oracle server.

Note: Interactive SQL sends a `CREATE TABLE xxx` command to Access Manager for each connection. Consequently, the job log contains the following message, which can be ignored:

```
ORA-00901: invalid CREATE command
```

8.1.2 Connecting from an AS/400 Precompiled Application

Three sample programs are distributed with Access Manager. They are `SAMP1C`, `SAMP1CBL`, and `SAMP1RPG`. Use the `CRTORAPKG` command to precompile, compile, and bind these programs. Refer to [Chapter 6, "Configuring Access Manager"](#) for more information about the `CRTORAPKG` command.

After using the `CRTORAPKG` command, you can execute the sample programs using the `CALL` command from the AS/400 command line. For example:

```
CALL SAMP1RPG
```

Each program explicitly connects to a server called `GENERIC` using an Oracle server user name of `SCOTT` and a password of `TIGER`. You can edit these sample programs if you want to test them against a different server or if you want to use a different user name or password.

After execution, you can issue a `SELECT` statement for the `DEPT` table through interactive SQL (`STRSQL`) to verify that the rows were inserted successfully.

These sample programs (included in [Appendix A, "Sample Programs"](#)) are in the Access Manager installation library in the following files:

Table 8–1 Sample Program Files

Program	File
SAMP1C	QSQLCSRC
SAMP1CBL	QCBLLSRC
SAMP1RPG	QRPGLESRC

8.2 Connection Management

Connection management consists of connecting to and disconnecting from one or more servers, and using `COMMIT` and `ROLLBACK` to coordinate transactions.

Access Manager connects to an Oracle server using the DB2/400 `CONNECT` syntax. The Oracle `CONNECT` syntax is not supported by Access Manager.

An application can connect to an Oracle server explicitly or implicitly. Explicit connection occurs when the DB2/400 `CONNECT TO` statement is issued. Implicit connection occurs when the first SQL statement in an application program is not a `CONNECT TO` statement. If this occurs, then the connection is made to the server that is specified on the `RDB` parameter in the `IBM CRTSQLxxx` command.

After a connection is established, the server becomes active, and all SQL statements are processed on that server.

Access Manager imposes no restrictions on the number of connections that can be opened at one time. In a single session, you can connect to multiple Oracle servers and multiple IBM DRDA application servers. When multiple servers are connected, you can switch between servers using the DB2/400 `SET CONNECTION` command.

For example:

```
CONNECT TO LOCAL
INSERT INTO DEPT VALUES (50, 'LOCAL', 'ROW 1')
CONNECT TO ORACLE1 USER SCOTT USING 'TIGER'
INSERT INTO DEPT VALUES (50, 'ORACLE1', 'ROW 1')
CONNECT TO ORACLE2 USER JOHN USING 'SMITH'
INSERT INTO DEPT VALUES (50, 'ORACLE2', 'ROW 1')
SET CONNECTION ORACLE1
INSERT INTO DEPT VALUES (50, 'ORACLE1', 'ROW 2')
SET CONNECTION LOCAL
INSERT INTO DEPT VALUES (50, 'LOCAL', 'ROW 2')
```

If multiple connections are open and a `COMMIT` or `ROLLBACK` statement is issued, then all database servers are requested to `COMMIT` or to `ROLLBACK`.

You cannot have multiple connections open to the same `RDB` name within a single session. For example, the following would be invalid:

```
CONNECT TO ORACLE1 USER X USING 'PWDX'
CONNECT TO ORACLE1 USER Y USING 'PWDY'
```

All applications, including interactive SQL (`STRSQL`), keep all connections open until you do one of the following:

- sign off from your AS/400 system
- issue a `RELEASE` command for each open connection, followed by a `COMMIT` statement
- issue a `RELEASE ALL` command, followed by a `COMMIT` statement, thereby ending all connections

Exiting Interactive SQL (`STRSQL`) does not affect the status of any connections that were obtained (and are still active) within Interactive SQL. All Oracle server connections that are left open are available on subsequent invocations of Interactive SQL within the same job. However, exiting a program that uses embedded SQL in the `EXEC SQL` format will cause any Oracle server connection that was created within that program to be released.

Note: The DB2/400 `RELEASE` command followed by a `ROLLBACK` statement does not end the connection.

Access Manager connects to an Oracle server using `CONNECT` type 5 and establishes a protected conversation, as documented in the IBM DB2/400 SQL Programming Guide.

Note: Because Access Manager runs as a protected conversation, an attempt to use the `DISCONNECT` command results in the following message for both TCP/IP and APPC/LU6.2 connections:

```
SQL0858 "Cannot disconnect relational database
<server_name> due to LU6.2 protected conversation"
```

8.3 Describing Tables

Interactive SQL (STRSQL) does not allow the `DESCRIBE TABLE` to be issued interactively. However, column and data type information can be obtained through interactive SQL (STRSQL) by using the prompt option (PF4) when issuing a `SELECT` from the command line. This information can be used by application programmers to understand how Oracle data types are mapped to AS/400 data types.

8.4 Stored Procedures

On the AS/400, you call a stored procedure by using the IBM DB2/400 `CALL` statement. The `CALL` statement can be executed interactively from interactive SQL or embedded in an application within the appropriate `EXEC-SQL END-EXEC` delimiters for the host language.

Access Manager allows access from AS/400 applications to Oracle stored procedures or packages by providing mapping from the IBM `CALL` statement to an Oracle stored procedure.

The results of a `CALL` command can be committed or rolled back. Access Manager always assumes that an invocation of stored procedures has changed something on the Oracle server.

8.4.1 Advantages of Stored Procedures

As defined on the AS/400, a stored procedure is a programming construct that can be called from within the SQL environment to perform a set of operations. The operations can include host language statements, SQL statements, and logic. When connected to an Oracle server, the host language statements are packaged within a PL/SQL stored procedure on the Oracle server.

Stored procedures in PL/SQL provide the same benefits as procedures in a host language such as C or COBOL. A PL/SQL procedure that is stored in the Oracle Database can be called from several programs. The use of stored procedures can also enhance the performance of a distributed application. For example, assume that you want to execute several SQL statements at an Oracle server using interactive SQL. If stored procedures are not used, then the AS/400 must send a separate request to the Oracle server for each SQL statement. If the same SQL statements are stored in a stored procedure at the Oracle server, then a single `CALL` statement on the AS/400 executes the entire block of SQL statements.

8.4.2 Stored Procedure Parameters

Parameters that are used only to pass data to a stored procedure are called `IN` parameters. Constant values that are used as parameters can be used only as `IN` parameters. Parameters that are used only to return data from a stored procedure are called `OUT` parameters. Parameters that can be used in both directions are called `IN OUT` parameters. You must use a host variable to have data returned from an Oracle stored procedure.

Calls to a stored procedure on the Oracle server from within an AS/400 application can use `IN`, `OUT` and `IN OUT` parameters. This enables parameters to be passed to the stored procedure and data to be returned. Calls to a stored procedure from within the interactive SQL environment can use only `IN` parameters. When using interactive SQL (STRSQL), you can pass constant parameters to the stored procedure on the Oracle server, but no data can be returned.

8.4.3 Declaring Procedures

When calling an Oracle stored procedure, you must use the DB2/400 `DECLARE PROCEDURE` statement. This statement is used to specify the `IN`, `OUT`, or `IN OUT` capability and the data type of each parameter. For example:

```
DECLARE Sample PROCEDURE (IN :arg1 INTEGER, IN OUT :arg2 CHAR(20),
                          OUT :arg3 SMALLINT) (LANGUAGE C SIMPLE CALL)
```

Caution: If a `DECLARE PROCEDURE` statement is not found for a `CALL` to a stored procedure, then each parameter defaults to `IN OUT`, and the data type for each parameter defaults to the data type of the associated host variable.

Ensure that the data types that are specified in the `DECLARE PROCEDURE` statement are compatible with those that are found in the stored procedure specification of the Oracle stored procedure. In addition, the input or output modes of each parameter (specified by `IN`, `OUT`, or `IN OUT` in the `DECLARE PROCEDURE` statement) and the number of parameters *must* match those of the Oracle stored procedure.

The following example is an extract from an AS/400 C program that calls the Oracle stored procedure `GETRESULTS`. The data types of parameters match, the input and output modes match, and the number of parameters match (nine parameters are in the `DECLARE PROCEDURE` statement on the AS/400, in the `CALL` to the procedure, and in the declaration of the Oracle stored procedure).

Example:

```
EXEC SQL BEGIN DECLARE SECTION;
char p1[10], p3[10], p5[10];
struct {short len; char value[20];} p1a; /* A VARCHAR */
struct {short len; char value[20];} p3a; /* A VARCHAR */
struct {short len; char value[20];} p5a; /* A VARCHAR */
long p2, p4, p6;
EXEC SQL END DECLARE SECTION;
. . . . .
EXEC SQL DECLARE GETRESULTS PROCEDURE
(IN :par1 CHAR(10), IN :par1a VARCHAR(20), IN :par2 INTEGER,
 INOUT :par3 CHAR(10), INOUT :par3a VARCHAR(20),
 INOUT :par4 INTEGER,
 OUT :par5 CHAR(10), OUT :par5a VARCHAR(20),
```

```

OUT :par6 INTEGER)
( LANGUAGE C SIMPLE CALL);
EXEC SQL CALL GETRESULTS(:p1, :p1a, :p2, :p3, :p3a, :p4, :p5, :p5a,
                        :p6);
. . . . .
The following is the file "getresults.sql" on the ORACLE server.
create or replace procedure GETRESULTS
(A1 IN char := 'DefA1', A1A IN VARCHAR2, A2 IN NUMBER,
 A3 IN OUT char, A3A IN OUT VARCHAR2, A4 IN OUT NUMBER,
 A5 OUT char, A5A OUT VARCHAR2, A6 OUT NUMBER)
IS
BEGIN
  A5 := A3; A5A := A3A; A6 := A4;
  A3 := A1; A3A := A1A; A4 := A2;
END;
/
~

```

8.5 Conversion of Oracle Data Types to DB2/400 Data Types

Oracle Access Manager for AS/400 converts Oracle data types to DB2/400 data types to allow application host variables to store and represent the Oracle data that they receive into a format that is native to the AS/400 environment.

The following table describes these data type conversions. In the table, *p* refers to precision, and *s* refers to scale. On DB2/400, VARCHAR can contain 32 740 characters at most.

Table 8–2 Oracle data types mapped to DB2/400 data types

Oracle Server Data Type	DB2/400 Data Type
CHAR	CHAR
DATE	DATE
LONG	VARCHAR
LONG RAW	VARCHAR
NUMBER (p, s)	SMALLINT 1<=p<=4, s=0 INTEGER 5<=p<=9, s=0 DECIMAL 10<=p<=31, s=0 or s<>0, p<=31 FLOAT anything else
RAW	VARCHAR
ROWID	VARCHAR
TIMESTAMP	TIMESTAMP
VARCHAR2	VARCHAR

LONG data types or combinations of LONG and LONG RAW data types are limited by the DB2/400 restriction that the entire length of data returned for a row must be less than 32 767 bytes. This restricts LONG and LONG RAW data types to a maximum of 32 740 bytes for all the LONG and LONG RAW data types that are returned in a row.

8.6 Message and Error Code Processing

Access Manager communicates all error conditions to the AS/400 application program through the SQLCA. Because Oracle error codes differ from DB2/400 error codes, a one-to-one mapping is not possible for all error conditions. Therefore, Access Manager uses the `SQLSTATE` field of the SQLCA as a mechanism to communicate Oracle errors to DB2/400.

When a `SQLSTATE` other than `00000` is returned from Access Manager, DB2/400 attempts to map the `SQLSTATE` to a DB2/400 `SQLCODE` and message. If it is unable to recognize the `SQLSTATE`, then DB2/400 issues the common message for that class code. For a list of `SQLSTATE` class codes, refer to the IBM DB2/400 SQL programming guide.

In some situations, Access Manager translates a `SQLSTATE` value into a valid DB2/400 `SQLCODE`. This is called a mapped error code. When mapping occurs, a valid DB2/400 `SQLCODE` is returned, and all applicable substitution tokens are placed in the `SQLERRMC` field of the SQLCA. This enables you to see valid text in the replacement variables of the DB2/400 SQL error message. Access Manager currently maps these error codes as indicated in the following table:

Table 8–3 Error Code Mapping

Oracle Error Code	DB2/400 SQLSTATE	DB2/400 SQLCODE
ORA-00901 invalid create command	42601	-104
ORA-00904 invalid column name	42703	-206
ORA-00911 invalid character	42601	-7
ORA-00923 from keyword not found	42601	-104
ORA-00932 inconsistent datatypes	42884	-440
ORA-00933 SQL command not properly ended	42601	-104
ORA-00936 missing expression	42601	-104
ORA-00942 table or view does not exist	42704	-204
ORA-01095 DML statement process 0 rows	02000	+100
ORA-01400 cannot insert NULL into string	23502	-407
ORA-01401 inserted value too large for column	22001	-404
ORA-01403 no data found	02000	+100
ORA-01405 fetched column value is NULL	22002	-305
ORA-01406 fetched column value was truncated	22001	-404
ORA-01422 exact fetch returns more than requested number of rows	21000	-811
ORA-01424 missing or illegal character following the escape character	22025	-130
ORA-01425 escape character must be character string of length 1	22019	-130
ORA-01427 single row subquery returns more than one row	21000	-811

Table 8–3 (Cont.) Error Code Mapping

Oracle Error Code	DB2/400 SQLSTATE	DB2/400 SQLCODE
ORA-01438 value larger than specified precision for this column	22001	-404
ORA-01455 converting column overflows integer datatype	22003	-304
ORA-01457 converting column overflows decimal datatype	22003	-304
ORA-01476 divisor is equal to zero	22012	-802
ORA-01479 last character in buffer is not NULL	22024	-302
ORA-01480 trailing null missing from STR bind variable	22024	-302
ORA-01488 invalid nibble or byte in input data	22023	-302
ORA-12154 TNS: could not resolve service name	08003	-30000
ORA-12533 TNS: illegal ADDRESS parameters	08003	-30000
ORA-12541 TNS: no listener	08003	-30000
ORA-12154 TNS: could not resolve service name	08003	-30000
ORA-12560 TNS: protocol adapter error	08003	-30000

If DB2/400 does not recognize a SQLSTATE, then the following message is issued:

```
SQL7940 SQLSTATE class unknown, SQLSTATE xxxxx...
```

Examine the job log for the relevant Oracle error that caused the SQLSTATE xxxxxx to be issued.

If Access Manager is unable to map an error code, then it returns a SQLCODE of -969 and a SQLSTATE which applies to the class code of the error. For example, all syntax errors have a SQLSTATE class code of 42. Therefore, if Access Manager receives a syntax parsing error from the Oracle server (which it cannot map), then it returns SQLCODE of -969 and SQLSTATE 42000.

If you are unable to determine the root cause of the error based on the SQLCODE and SQLSTATE that are returned to the application, then check the job log for more details. All Oracle errors and message text are displayed in the job log at the time that the error occurs.

Error Diagnosis and Reporting

This chapter discusses error diagnosis and reporting for Oracle Access Manager for AS/400. It includes the following sections:

- [Trace Files from the Access Manager](#) on page 9-1
- [Trace Files for Oracle Net](#) on page 9-2
- [Contacting Oracle Support Services](#) on page 9-2
- [Error categories](#) on page 9-2
- [How to Provide Items for Problem Diagnosis](#) on page 9-3

9.1 Trace Files from the Access Manager

Trace files are emitted by the Access Manager according to parameters set on the CHGSQLDBG screen. To cause a trace file to be written, place a 1 on the line titled "Access Manager debugging control string." Members will be placed in the Access Manager trace file which is designated by the next two lines on the CHGSQLDBG screen. If you need to create a new trace file, then use the following command as a template:

```
CRTPF mylib/mytracefile RCDLEN(240) MAXMBRS(*NOMAX)
```

where *mylib* is any library and *mytracefile* is any file name you choose. Update the CHGSQLDBG panel when changing the name and/or library where the trace files are to be written. You should give enough authority to users of the Access Manager so that this file can be written.

Members written to the trace file will be named similarly to MBRnnnnnnn where *nnnnnn* will start at 000001 for each Access Manager session. One can avoid the resetting of the Trace File member sequence numbers by using the following:

1. Turn on tracing through the CHGSQLDBG command.
2. Start an Access Manager session.
3. Check the QTEMP library for a User Space named ARDTRCSER.
4. Move this object to the Access Manager library.
5. Ensure that you give enough authority to Access Manager users so they can read and write this User Space.

9.2 Trace Files for Oracle Net

If so specified on the `CHGSQLNET` command, then Oracle Net trace files will be written to a file named `instance/TRC` where "instance" is the library of the Access Manager install. Members in this file will have names similar to `CLNTnnnnnn`, where the `nnnnnn` is the job number of the Access Manager job.

9.3 Contacting Oracle Support Services

To maximize the effectiveness of your queries, gather the following information before contacting Oracle Support Services:

- OS/400 release number
- Exact Access Manager release number (found in the JobLog)
- Exact Oracle Net release number (found in the JobLog)
- Oracle Database Server release number and platform
- You may be asked to supply a JobLog and trace files for problem diagnosis. See [Section 9.5, "How to Provide Items for Problem Diagnosis"](#) for information about how to provide these items.

9.4 Error categories

Use the following error categories to describe the error:

- [Documentation Error](#)
- [Incorrect Output](#)
- [Oracle External Error](#)
- [Performance](#)
- [Missing Functionality](#)

9.4.1 Documentation Error

When reporting documentation errors, you will be asked to provide the

- document name
- document part number
- document publication date
- page number

Describe the error in detail. Documentation errors can include erroneous documentation and omission of required information.

9.4.2 Incorrect Output

In general, an incorrect output error occurs whenever an Oracle supplied program (such as the Access Manager) produces a result that differs from what would be expected. When describing errors of this type, you need to describe, in detail, exactly what you expect the output to be; any non-obvious reason why you expect different output should be explained in detail.

If you think that you have found a software bug, then be prepared to answer the following questions:

- What are the exact SQL statements that cause the problem, if it is a data error.
- What is the full OS/400 Version in which the error occurs; what is the version of the Access Manager, CORE, and Oracle Net (the Oracle information should appear in the JobLog)
- What exactly is the problem and how is it reproduced

Try to pare down the number of steps or the program that causes the error; the simpler the number of steps that Oracle support needs to re-create the problem, the better; the simpler the user program that causes the error, the better.

9.4.3 Oracle External Error

Most Oracle errors (and some warnings) are placed in the JobLog.

9.4.4 Performance

AS/400 system performance is determined by many factors, most of which are not in the control of Oracle. Considerations such as system load, I/O topology, network topology, and DB2/400 resource availability make documentation of performance errors difficult.

Provide detailed information about the state of your environment when reporting a performance problem. Specific documentation might include:

- CPU type and memory configuration
- network topology
- Oracle workload characteristics
- query execution plans

9.4.5 Missing Functionality

Enhancement requests can be opened with Oracle Support Services to request the inclusion of functions and features that are not currently available with Oracle products such as the Access Manager. When opening an enhancement request, describe the specific feature or function to be added to the Access Manager, and provide a business case to justify the enhancement.

9.5 How to Provide Items for Problem Diagnosis

Here are some of the items that should be provided for problem diagnosis:

- A JobLog that may show the error. Even if it does not show the error explicitly, Support Services may be able to use the JobLog for problem determination and resolution. The Job Log should be provided as a member of a physical file that has a maximum record length of 133. Use the CPYSPLF command to copy a spooled JobLog to a physical file. Use a member name of JBLGkkkkkk where kkkkkk is the Job Number.
- An Access Manager trace file. This should be provided as an actual physical file. If you want to provide the actual physical file used by the Access Manager, then specify which member contains the trace to be perused.
- An Oracle Net trace file. This should be provided as an actual physical file. If you want to provide the actual physical file, please tell us the job number of the job that had the problem.

- If the error occurs in STRSQL, then use PF13 and then option 2 to print a spooled copy of the session log. Use CPYSPLF to copy this spooled file to the same physical file to which the JobLog was copied. You should use a member named of STRSQL $mmmm$.
- If the error occurred in a user program, try to pare down the program to a point where the error still occurs. If possible, try to pare the program down to a point where "include" files are not needed. Provide a physical source file containing the source of the program; include the "include" files if they are needed to compile the program.
- Any other files that you feel would help Support Services diagnose the problem are appreciated. Add a note as to why these additional files are relevant.
- Include a README file which exactly specifies the problem. Tell us if the problem was not apparent on previous releases of the Access Manager. Tell us if the problem was not apparent when using previous versions of Oracle. We need to know the version numbers (all five fields as in 10.2.0.1.0). Tell us if the configuration of your AS/400 has recently changed. This includes changing the system name, and reconfiguration of TCP/IP addresses.

Note: The "timestamps" on the files provided, especially for the JobLogs and Trace Files, should be consistent. They should all be produced at about the same time. Including a JobLog from a Sunday run and trace files from a Tuesday run is not a good idea. If you submit files with inconsistent time stamps, then you will most likely be asked to gather the information again.

Package up all the files into a Save File. Support services will tell you where to FTP such a Save File. Remember to FTP the Save File in binary mode. Submission of files in PDF or MicroSoft Word format is strongly discouraged.

Sample Programs

This appendix includes sample Access Manager programs that are written in C, COBOL, and RPG. The following sample programs are included:

- [Sample Access Manager C Program](#) on page A-1
- [Sample Access Manager COBOL Program](#) on page A-3
- [Sample Access Manager RPG Program](#) on page A-4

A.1 Sample Access Manager C Program

```

#include <string.h>
#include <stdlib.h>
#include <stdio.h>
/*-----*/
/*                                          */
/* Program Name:  SAMP1C                    */
/*                                          */
/* Function: Insert a sample row into the DEPT table */
/*                                          */
/* Author:   Oracle Corporation            */
/* Date:    8/25/95                       */
/*                                          */
/*-----*/
#ifdef TRUE
# undef TRUE
#endif
#define TRUE 1

EXEC SQL INCLUDE sqlca;

struct sqlca *sca;

main()
{

EXEC SQL BEGIN DECLARE SECTION;
  char server??(10??);
  char username??(10??);
  char password??(10??);
  long int deptno;
  char dname??(14??);
  char loc??(13??);
  char sql_stmt??(80??);

```

```

EXEC SQL END DECLARE SECTION;

long int rc;
EXEC SQL WHENEVER SQLERROR GOTO big_error;
EXEC SQL WHENEVER SQLWARNING GOTO big_warn;

printf("Entering test SAMP1C...\n");
/*
strcpy(server, "GENERIC");
strcpy(username, "SCOTT");
strcpy(password, "TIGER");
EXEC SQL CONNECT TO :server USER :username USING :password;
*/
printf("After implicit connect.\n");

strcpy(sql_stmt,
"INSERT INTO DEPT VALUES (?, ?, ?)");

EXEC SQL PREPARE S FROM :sql_stmt;
printf("After prepare.\n");

deptno = 88;
strcpy(dname, "SAMP1 C");
strcpy(loc, "Success");

EXEC SQL EXECUTE S USING :deptno, :dname, :loc;
printf("After execute.\n");

EXEC SQL RELEASE CURRENT;
printf("After release current.\n");

EXEC SQL COMMIT;
printf("After commit.\n");
printf("Exiting test SAMP1C...\n");

exit(0);

big_warn:
EXEC SQL WHENEVER SQLWARNING CONTINUE;
printf("Big warn\n");
big_error:
EXEC SQL WHENEVER SQLERROR CONTINUE;
printf("Big error!: SQLCA...\n");
printf("      sqlcaid  =%0.8s.\n", sqlca.sqlcaid);
printf("      sqlcabc  =%d.\n", sqlca.sqlcabc);
printf("      sqlcode  =%d.\n", sqlca.sqlcode);
printf("      sqlerrml =%d.\n", sqlca.sqlerrml);
printf("      sqlerrmc =%0.70s.\n", sqlca.sqlerrmc);
printf("      sqlerrp  =%0.8s.\n", sqlca.sqlerrp);
printf("      sqlerrd  =%d %d %d %d %d %d\n", sqlca.sqlerrd??(0??),
      sqlca.sqlerrd??(1??), sqlca.sqlerrd??(2??),
      sqlca.sqlerrd??(3??), sqlca.sqlerrd??(4??),
      sqlca.sqlerrd??(5??));
printf("      sqlwarn  =%0.11s.\n", sqlca.sqlwarn);
printf("      sqlstate =%0.5s.\n", sqlca.sqlstate);
exit(1);
not_found:
exit(0);
}

```

A.2 Sample Access Manager COBOL Program

```
IDENTIFICATION DIVISION.
PROGRAM-ID. SAMP1CBL.
    AUTHOR. Oracle.
    INSTALLATION. Oracle Corporation.
    DATE-WRITTEN. August 25, 1995.
    DATE-COMPILED.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
    SOURCE-COMPUTER. IBM-AS400.
    OBJECT-COMPUTER. IBM-AS400.
DATA DIVISION.
WORKING-STORAGE SECTION.
    EXEC SQL BEGIN DECLARE SECTION END-EXEC.
77 SERVER-NAME PIC X(10).
77 USER-NAME PIC X(10).
77 USER-PASSWORD PIC X(10).
* Anything going into a NUMERIC, DECIMAL, SMALLINT, INTEGER
* column MUST have a sign.
* DEPTNO in the following is a four-byte zoned decimal number
* with the sign in the "zone" portion of the right hand character
77 DEPNO PIC S9(4).
77 DEPTNAME PIC X(14).
77 MISC PIC X(13).
77 SQL-STATEMENT PIC X(100).
    EXEC SQL END DECLARE SECTION END-EXEC.
    EXEC SQL INCLUDE SQLCA END-EXEC.
77 CONV1 PIC S999999999 SIGN IS LEADING SEPARATE.
77 CONV2 PIC S999999999 SIGN IS LEADING SEPARATE.
77 CONV3 PIC S999999999 SIGN IS LEADING SEPARATE.
PROCEDURE DIVISION.
P1NTVLE-INIT.
    MOVE "GENERIC" TO SERVER-NAME.
    MOVE "SCOTT" TO USER-NAME.
    MOVE "TIGER" TO USER-PASSWORD.
    MOVE SPACES TO SQLERRMC.
    EXEC SQL WHENEVER SQLERROR GO TO BIG-ERROR END-EXEC.
    EXEC SQL WHENEVER SQLWARNING GO TO BIG-WARNING END-EXEC.
P1NTVLE-CONNECT.
    DISPLAY "Connected implicitly...".
P1NTVLE-PREPARE.
    MOVE "Insert into DEPT VALUES(?,?,?)" to SQL-STATEMENT.
    DISPLAY " " SQL-STATEMENT.
    EXEC SQL PREPARE S FROM :SQL-STATEMENT END-EXEC.
    DISPLAY "Prepared ...".
P1NTVLE-EXECUTE.
    MOVE 88 TO DEPNO.
    MOVE "SAMP1 CBL" TO DEPTNAME.
    MOVE "Success" to MISC.
    EXEC SQL EXECUTE S USING :DEPNO, :DEPTNAME, :MISC END-EXEC.
    DISPLAY "Executed...".
P1LE-RELEASE.
    EXEC SQL RELEASE CURRENT END-EXEC.
    DISPLAY "Released current...".
P1LE-COMMIT.
    EXEC SQL COMMIT END-EXEC.
    DISPLAY "Committing ...".
P1LE-END.
    DISPLAY "Exiting ...".
```

```

        STOP RUN.
BIG-ERROR.
        Display "SQL Error ...".
        PERFORM DUMP-SQLCA.
        GO TO P1LE-END.
BIG-WARNING.
        Display "SQL Warning ...".
        PERFORM DUMP-SQLCA.
        GO TO P1LE-END.
DUMP-SQLCA.
*Move binary fields to displayable fields for DISPLAY stmt.
        MOVE SQLCODE TO CONV1. MOVE SQLSTATE TO CONV2.
        DISPLAY " SQLCODE=" CONV1 ", SQLSTATE=" CONV2.
        DISPLAY " SQLERRMC on following line ...".
        DISPLAY " " SQLERRMC.
        DISPLAY " SQLERRP=" SQLERRP.
        MOVE SQLERRD(1) TO CONV1.
        MOVE SQLERRD(2) TO CONV2.
        MOVE SQLERRD(3) TO CONV3"
        DISPLAY "SQLERRD(1)..SQLERRD(3)=" CONV1 " " CONV2
                " " CONV3.

        MOVE SQLERRD(4) TO CONV1.
        MOVE SQLERRD(5) TO CONV2.
        MOVE SQLERRD(6) TO CONV3.
        DISPLAY "SQLERRD(4)..SQLERRD(6)=" CONV1 " " CONV2
                " " CONV3.

        DISPLAY "SQLWARN=" SQLWARN.

```

A.3 Sample Access Manager RPG Program

```

H
F* FILE DECLARATION FOR QPRINT
FQPRINT   O   F 132           PRINTER
DVAR      DS
DDEPTNO           1           4b 0
DHOSTDN          5           8b 0
DSERVER          9           18
DUSERNM         19           28
DPASSWD         29           38
DSTMSQL         39           118
DSTR           119           198   DIM(80)
DSTMT1          C           'Insert into DEPT -
D                VALUES(88,'SAMP1 RPG','Success')'
C                MOVE      *BLANKS      STR
C                CLEAR      SQLCA
C*PUT OUT "ENTERING TEST P1..." MESSAGE
C                MOVE(L(P) 'GENERIC'     SERVER
C                MOVE(L(P) 'SCOTT'       USERNM
C                MOVE(L(P) 'TIGER'       PASSWD
C                MOVE(L(P) STMT1         STMSQL
C                EXCEPT NTRP1
C                EXCEPT NTRP2
C/EXEC SQL
C+  WHENEVER SQLERROR GOTO BIGERR
C/END-EXEC
C/EXEC SQL
C+  WHENEVER SQLWARNING GOTO BIGWRN
C/END-EXEC
C*EXEC SQL

```



```

C*   CONNECT TO :SERVER USER :USERNM USING :PASSWD
C*   CONNECT
C*END-EXEC
C* PUT OUT "CONNECTED IMPLICITLY" MESSAGE...
C           EXCEPT   CNCTD
C/EXEC SQL PREPARE S FROM :STMSQL
C/END-EXEC
C* PUT OUT "AFTER PREPARE" MESSAGE
C           EXCEPT   PRPAF
C* NOW DO THE EXECUTE OF THE SQL STATEMENT AND PUT OUT MESSAGE
C/EXEC SQL EXECUTE S
C/END-EXEC
C           EXCEPT   XEQAF
C* NOW DO THE RELEASE OF THE SERVER AND PUT OUT MESSAGE
C/EXEC SQL RELEASE CURRENT
C/END-EXEC
C           EXCEPT   RLSAF
C* NOW DO THE COMMIT AND PUT OUT MESSAGE ...
C/EXEC SQL COMMIT
C/END-EXEC
C           EXCEPT   CMTAF
C* AND NOW WE ARE GOING TO EXIT ...
C           EXCEPT   XITNW
C           GOTO       FINISH
C*
C   BIGWRN      TAG
C/EXEC SQL WHENEVER SQLWARNING CONTINUE
C/END-EXEC
C           EXCEPT   WRNMSG
C           EXSR       PSQCA
C           GOTO       FINISH
C*
C   BIGERR      TAG
C/EXEC SQL WHENEVER SQLEERROR CONTINUE
C/END-EXEC
C           EXCEPT   ERRMSG
C           EXSR       PSQCA
C           GOTO       FINISH
C*
C   FINISH      TAG
C           SETON
C*
CSR   PSQCA     BEGSR
C           EXCEPT   SQLCA1
C           EXCEPT   SQLCA2
C           EXCEPT   SQLCA3
C           EXCEPT   SQLCA4
C           EXCEPT   SQLCA5
C           EXCEPT   SQLCA6
CSR   ENDSR
C*
OQPRINT  E           NTRP1      1
O
O           E           NTRP2      1
O
O           STMSQL
O           E           CNCTD      1
O
O           E           PRPAF      1
O

```

LR

7 'STMSQL='

'Entering SAMP1RPG...'

'CONNECTED IMPLICITLY'

'AFTER PREPARE'

```

0      E      XEQAF      1      'AFTER EXECUTE'
0
0      E      RLSAF      1      'AFTER RELEASE CURRENT'
0
0      E      CMTAF      1      'AFTER COMMIT'
0
0      E      XITNW      1      'EXITING SAMP1 RPG'
0
0      E      WRNMSG      1      'BIG WARN:SQLCA...'
0
0      E      ERRMSG      1      'BIG ERROR:SQLCA...'
0
0      E      SQLCA1      1
0      8 ' SQLAID='
0      SQLAID      +0
0      +0 ', SQLABC='
0      SQLABC      +0
0      +0 ', SQLCOD='
0      SQLCOD      +0L
0      +0 ', SQLERL='
0      SQLERL      +0
0      E      SQLCA2      1
0      8 ' SQLERM='
0      SQLERM      +0
0      E      SQLCA3      1
0      8 ' SQLERP='
0      SQLERP      +0
0      E      SQLCA4      1
0      16 ' SQLER1..SQLER3='
0      SQLER1      +0L
0      SQLER2      +1L
0      SQLER3      +1L
0      E      SQLCA5      1
0      16 ' SQLER4..SQLER6='
0      SQLER4      +0L
0      SQLER5      +1L
0      SQLER6      +1L
0      E      SQLCA6      1
0      8 ' SQLWRN='
0      SQLWRN      +0
0      +0 ', SQLSTT='
0      SQLSTT      +0

```

Globalization Support

This appendix documents the Globalization Support for the Access Manager. The following topics are included:

- [Access Manager Globalization Support](#) on page B-1
- [Oracle Character Set Names](#) on page B-1
- [Oracle NLS_LANG Specifications](#) on page B-2

B.1 Access Manager Globalization Support

The Access Manager is designed to support a multitude of single-byte character sets. Source files containing SQL packages can be written in any of the coded character set identifiers (CCSID) that are listed in [Table B-1](#). The Access Manager does not support double-byte character sets as the underlying IBM ARD interface does not support those character sets.

B.2 Oracle Character Set Names

In the table are the Oracle character set names that can be included as the third part of an Oracle Globalization Support language specification. The Oracle language specification can be explicitly set for an invocation of the Access Manager by having access to a data area named NLS_LANG.

Table B-1 Oracle Character Set Names

CCSID	Oracle Character Set Name
37	WE8EBCDIC37
273	D8EBCDIC273
277	DKEBCDIC277
278	S8EBCDIC278
280	I8EBCDIC280
284	WE8EBCDIC284
285	WE8EBCDIC285
297	F8EBCDIC297
500	WE8EBCDIC500
838	TH8TISEBCDIC

Table B-1 (Cont.) Oracle Character Set Names

CCSID	Oracle Character Set Name
870	EE8EBCDIC870
875	EL8EBCDIC875
1140	WE8EBCDIC1140
1141	D8EBCDIC1141
1142	DK8EBCDIC1142
1143	S8EBCDIC1143
1144	I8EBCDIC1144
1145	WE8EBCDIC1145
1146	WE8EBCDIC1146
1147	F8EBCDIC1147
1148	WE8EBCDIC1148

The NLS_LANG data area that is shipped with the Access Manager defaults to AMERICAN_AMERICA.WE8EBCDIC37. All data emanating from a SQL package is data to be transmitted from the AS/400 to the Oracle Database. This data is translated from the character set that is associated with the CCSID of the package to the character set that is denoted explicitly by NLS_LANG or by the default for NLS_LANG. Data that is returned from the Oracle server to the AS/400 is translated by the IBM Application Requester Driver API.

B.3 Oracle NLS_LANG Specifications

An Oracle NLS_LANG specification includes three parts:

1. a language
2. a territory
3. a character set

For example:

```
language_territory.character_set
AMERICAN_AMERICA.D8EBCDIC1141
```

For the Access Manager, the language is AMERICAN, and the territory is AMERICA. The error messages are currently in American English only. The third part is the Oracle character set name from [Table B-1](#). The NLS_LANG values in [Table B-2](#) are currently supported:

Table B-2 Oracle NLS_LANG Values

Language	Setting for language_territory.charset
Austrian/German	AMERICAN_AMERICA.D8EBCDIC273 and AMERICAN_AMERICA.D8EBCDIC1141
Danish/Norwegian	AMERICAN_AMERICA.DK8EBCDIC277 and AMERICAN_AMERICA.DK8EBCDIC1142
Eastern European	AMERICAN_AMERICA.EE8EBCDIC870

Table B-2 (Cont.) Oracle NLS_LANG Values

Language	Setting for language_territory.charset
Finnish/Swedish	AMERICAN_AMERICA.S8EBCDIC278 and AMERICAN_AMERICA S8EBCDIC1143
French	AMERICAN_AMERICA.F8EBCDIC297 and AMERICAN_AMERICA F8EBCDIC1147
Greek	AMERICAN_AMERICA.EL8EBCDIC875
Italian	AMERICAN_AMERICA.I8EBCDIC280 and AMERICAN_AMERICA I8EBCDIC1144
Thai	AMERICAN_AMERICA.TH8TISEBCDIC
Western European	AMERICAN_AMERICA.WE8EBCDIC37 and AMERICAN_AMERICA WE8EBCDIC1140
Western European	AMERICAN_AMERICA.WE8EBCDIC500 and AMERICAN_AMERICA WE8EBCDIC1148

Each user can have its own NLS_LANG data area. To change the NLS_LANG data area, change the Access Manager Globalization Support language parameter by using the CHGSQLDFLT command as described in [Chapter 6, "Configuring Access Manager"](#).

Symbols

- 16, valid AS/400 "Client trace level", 6-6
- *ADMIN, valid AS/400 "Client trace level", 6-6
- *ARDPGM
 - relational database directory entry in ADDRDBDIRE Panel 1, 5-4
 - remote location entry in configuration example, 5-3
- *BATCH, example "Batch/Terminal Compile" on Oracle SQL Package Panel, 6-8
- *C, example "Language Type" on Oracle SQL Package Panel, 6-8
- *CALLER, used in ACTGRP(*CALLER), for invoking Access Manager program, 1-6
- *COBOL, example "Language Type" on Oracle SQL Package Panel, 6-8
- *CURRENT, example "Target Release" on Oracle SQL Package Panel, 6-8
- *DUW with ROBCNNMTH parameter, known restriction, 2-4
- *FIRST
 - adding Access Manager library to library list, 6-1
 - with Access Manager installation library, 4-4
- *FLOAT
 - known problem with data type, 2-3
 - specifies floating point notation using default precision and scale values, 6-4
- *KEEP, setting Commitment-Control Resource Release, example, 6-5
- *New, example "Activation Group" on Oracle SQL Package Panel, 6-8
- *OFF, valid AS/400 "Client trace level", 6-6
- *PGM, example "Object Type" on Oracle SQL Package Panel, 6-8
- *PKDEC
 - known problem with data type, 2-3
 - specifies packed decimal notation using default precision and scale values, 6-4
- *RELEASE, setting Commitment-Control Resource Release, example, 6-5
- *RPG, example "Language Type" on Oracle SQL Package Panel, 6-8
- *SECADM special authority, logging on before installing Oracle Access Manager, 4-2
- *USER, valid AS/400 "Client trace level", 6-6

A

- Access Manager
 - see Oracle Access Manager
- Access Manager Commands Menu panel, 6-2
- Access Manager Debugging Options Panel
 - Panel (1), 6-6
 - Panel (2), 6-7
- Access Manager Default Options Panel
 - library, 6-2
 - user ID and password, 6-3
- Access Manager Network Parameters Panel
 - Panel (1), 6-5
 - Panel (2), 6-5
- activation groups
 - main topic, description, 1-6
 - ACTGRP(*CALLER) creates Access Manager, 1-6
 - commands
 - CRTSQLxxx, 1-6
 - STRSQL, 1-6
- ADD command, adds a server to the list of available servers, 7-10
- ADDLIB command, add Access Manager library to library list
 - configuration, 6-1
 - installation, 4-4
- ADDRDBDIRE command
 - adding an entry in RDB directory, 5-6
 - defining IBM DRDA connection, 5-2
- ADDRDBDIRE Panel_1, 5-4
- ADDRDBDIRE Panel_2, 5-5
- addressing protection, provided by an activation group, 1-6
- AM400_COM program
 - defined, 7-1
 - commands
 - main topic, 7-9
 - ADD, 7-10
 - FORCE, 7-11
 - LIST, 7-11
 - MODIFY, 7-10
 - QUERY, 7-10
 - REMOVE, 7-10
 - SET, 7-11
 - SHUTDOWN, 7-10
 - STARTUP, 7-10

- STATUS, 7-9
 - privileged ID field, 7-8
 - privileged user IDs, 7-7
 - with LIST command, 7-11
- AM400_RID program
 - defined, 7-1
 - altering the environment, 7-9
 - starting the program, 7-8
 - startup parameter, 7-7
 - with DATAQUEUE_TIMEOUT parameter, 7-6
 - with LIST command, 7-11
 - with resolve-in-doubt transactions, 7-3
 - with two-phase commit, 7-3
- Application Requester Driver (ARD) interface
 - data from Oracle server is translated by IBM ARD API, B-2
 - Oracle Access Manager architecture, 1-2
 - Oracle Access Manager component interactions, 1-3
- architecture of Oracle Access Manager
 - component interactions, 1-3
 - main topic, 1-2
- ARD, Application Requester Driver
 - Access Manager architecture, 1-2
 - data from Oracle server is translated by IBM ARD API, B-2
- ARD, IBM ARD, distributed database environment, 1-5
- AS/400 and OS/400, platform names for IBM eServer iSeries servers, viii
- AS/400 application, Access Manager component interactions, 1-3
- AVG function, precision and scale, 6-4

B

- back up any existing Access Manager library before upgrading to a new release, 4-2
- blocksize data area, default block size, 6-4
- bugs fixed as of this release, 2-1

C

- CALL command
 - code example, connecting from an AS/400 precompiled application, 8-2
 - declaring procedures with call to a stored procedure, 8-5
 - stored procedures, results can be committed or rolled back, 8-4
- CALL statement
 - calling a stored procedure by using IBM DB2/400, 8-4
 - single CALL statement on AS/400 executes entire block of SQL statements, 8-4

- change
 - Change or Enter New Server Attributes - panel, 7-6
 - Change Privileged IDs (Relative to AM400_COM) - panel, 7-8
 - Change RID (Resolve In Doubt) Options - panel, 7-5
 - Change RID (Resolve In Doubt) Options Panel, 6-9
 - changes and enhancements, 2-1
- CHAR data type, Oracle data types mapped to DB2/400 data types, 8-6
- checklists
 - enabling two-phase commit, 7-4
 - installation, 4-2
 - Oracle Net, TCP/IP AS/400 configuration, 5-4
 - post-installation, 4-4
- CHG2PCPRM command
 - main topic, changing two-phase commit parameters, 6-9
 - changing AM400_RID Options, 7-5
 - changing RID (Resolve In Doubt) Privileged IDs - panel, 7-7
 - purpose of command, 6-2
- CHGSQLDBG command
 - main topic, opens Access Manager Debugging Options - panel, 6-6
 - purpose of command, 6-2
- CHGSQLDFLT command
 - changing Access Manager Globalization Support language parameter, B-3
 - changing Oracle server default user ID and password, when using STRSQL, 8-2
 - changing two-phase commit parameter, 7-1
 - opening Access Manager Default Options - panel, 6-2
 - purpose of command, 6-2
- CHGSQLNET command
 - Access Manager commands, 6-2
 - network parameters panel, 6-5
- CHGUSRPRF command
 - changing default password during installation, 4-4
- CHKSQDFLT command, checking STRSQL default data type, precision, and scale values, 2-2
- clause, FOR FETCH ONLY, required for block fetching to be performed, 6-4
- CLEANUP_DELAY parameter
 - AM400_COM program can be used to dynamically alter environment of AM400_RID program, 7-9
 - AM400_COM QUERY command can query the value of this parameter, 7-10
 - description, a tuning parameter requiring a single numeric value (in seconds), 7-6
 - tuning parameter, Change RID (Resolve In Doubt) Options - panel, 7-6
- code example
 - accessing the AM400_COM program, 7-9
 - adding Access Manager library to your library list

- configuring, before using Access Manager commands, 6-1
- post installation of Access Manager, 4-4
- AM400_COM commands, 7-9
- an extract from an AS/400 C program that calls the Oracle stored procedure GETRESULTS, 8-5
- changing password after product installation, 4-4
- completing installation, 4-3
- connecting to the Oracle server with STRSQL, 8-1
- declaring procedures when calling an Oracle stored procedure, 8-5
- editing member TNSNAMES in the ORA file, 5-5
- instance name of Oracle server, 5-6
- invalid commands for multiple connections to same RDB name, 8-3
- LODRUN command to start installation, 4-2
- member names, file name mapping, 5-2
- message issued if DB2/400 does not recognize a SQLSTATE, 8-8
- precision and scale values from SQL statements, 6-4
- relationship between Access Manager and AS/400 TCP/IP configuration, 5-3
- sample Access Manager C program (Appendix A), A-1
- sample Access Manager COBOL program (Appendix A), A-3
- sample Access Manager RPG program (Appendix A), A-4
- SQL0858, protected conversation, 8-4
- starting a subsystem to enable two-phase commit processing, 7-8
- switching between servers using the DB2/400 SET CONNECTION command, 8-3
- text format conventions, examples of code input and output, ix
- using CHKSQDFLT command to check default data type, precision, and scale values, 2-2
- commands
 - activation groups, CRTSQLxxx, 1-6
 - ADDLIB, before using any Access Manager configuration utilities, 4-4
 - AM400_COM program
 - main topic, 7-9
 - AM400_COM program, description and code example
 - ADD, 7-10
 - FORCE, 7-11
 - LIST, 7-11
 - MODIFY, 7-10
 - QUERY, 7-10
 - REMOVE, 7-10
 - SET, 7-11
 - SHUTDOWN, 7-10
 - STARTUP, 7-10
 - STATUS, 7-9
 - CHG2PCPRM
 - before accessing Change RID Privileged IDs - panel, 7-7
 - changing AM400_RID options, 7-5
 - CHGSQDFLT, changing two-phase commit parameter, 7-1
 - CHGUSRPRF
 - changing default password, 4-4
 - CHKSQDFLT, known problems, checking default data type precision and scale values, 2-2
 - COMMIT
 - ending a unit of work (SQL commands), 1-6
 - known restriction, 2-3
 - Oracle required privileges, 7-9
 - prepare phase of two-phase commit transaction, 7-2
 - when AS/400 receives a prepared to commit message, 7-3
 - COMMIT HOLD, known restriction, 2-4
 - CREATE TABLE, job log message, ORA-00901, 8-2
 - CRTORAPKG command, failure due to severity level, 6-9
 - CRTSQLxxx
 - calling and running Access Manager, 1-6
 - known restriction with RDBCNNMTH parameter, 2-4
 - DB2/400
 - CONNECT TO, STRSQL to connect to the Oracle server, 8-2
 - SET CONNECTION, switching between multiple servers, 8-3
 - DSPJOBLOG, verifying installation, 4-3
 - EDTF, defining TNSNAMES entry, 5-5
 - GO CMDORAAM, displays Access Manager commands menu, 6-2
 - GRANT, granting privileges to AM400_RID program, 7-9
 - IBM
 - CALL, redirecting a procedure with Oracle SQL, 1-4
 - CRTSQL, RDB parameter and connection management, 8-3
 - LODRUN, starting the installation process, 4-2
 - menu, Access Manager Commands Menu, 6-2
 - menu, use GO CMDORAAM to display Access Manager commands menu, 6-2
 - Oracle Access Manager
 - main topic, 6-1
 - ADDLIB, 6-1
 - CALL, 8-2, 8-4
 - CHG2PCPRM, 6-9
 - CHGSQDDBG, 6-6
 - CHGSQDFLT, 6-2, 8-2, B-3
 - COMMIT, 6-5
 - CRTORAPKG, 6-7, 8-2
 - CRTSQLxxx, 6-3
 - DESCRIBE TABLE, 8-4
 - DISCONNECT, 8-4
 - purposes of commands, 6-2
 - RELEASE ALL, 8-3
 - ROLLBACK, 6-5

- SELECT, 8-4
- RELEASE, connection management, 8-3
- ROLLBACK
 - ending a unit of work, 1-6
 - if one site is not prepared to commit, 7-3
 - known restriction, 2-3
 - Oracle required privileges, 7-9
- SQL*Net
 - ADDRDBDIRE, adding an entry in the RDB directory, 5-6
 - ADDRDBDIRE, defines IBM DRDA connection, 5-2
 - WRKMBRPDM, editing member TNSNAMES in the ORA file, 5-5
- STRSBS, enabling two-phase commit processing, 7-8
- STRSQL, activation groups, 1-6
- COMMIT command
 - commitment control resource, 6-5
 - ending a unit of work (SQL commands), 1-6
 - known restriction, 2-3
 - Oracle required privileges, 7-9
 - prepare phase of two-phase commit transaction, 7-2
 - when AS/400 receives a prepared to commit message, 7-3
- COMMIT HOLD command, known restriction, 2-4
- commit phase, defined, 7-2
- commit point site, defined, 7-2
- COMMIT statement, connection management, 8-3
- commitment control resource, 6-5
- components, Oracle Access Manager, table of interactions, 1-3
- configuration parameters, test connectivity before changing any configuration parameters, 4-5
- configuring
 - Oracle Access Manager, 6-1
 - SQL*Net TCP/IP
 - configuring your client computers for TCP/IP, 5-4
 - edit member TNSNAMES in the ORA file, 5-5
- CONNECT
 - CONNECT SQL statement, known restriction, 2-4
 - CONNECT TO command, IBM DB2/400
 - command to connect to the Oracle server, 8-2
 - CONNECT TO statement, explicit connection to an Oracle server, 8-3
 - Oracle CONNECT syntax is not supported by Oracle Access Manager, 8-3

- connecting
 - connect processing with Oracle Access Manager, 1-5
 - connection management, 8-3
 - multiple connections with different RDB names, 1-6
 - multiple connections with same Oracle server, 1-6
 - test connectivity before changing any configuration parameters, 4-5
 - through Oracle Net, 5-2
 - through TCP/IP, 5-3
 - to Oracle server from precompiled application, 8-2
 - uniquely identifying a connection, 1-6
- conventions, text format for examples of code input and output, ix
- COUNT(*) function, returns NUMBER values with precision and scale set to zero, 6-4
- CREATE statement, a DDL statement, if two-phase commit is set to *NO, 6-4
- CREATE TABLE command, job log messages, 8-2
- CRTORAPKG command
 - caution, ensure that your Oracle server and your network connection are active before running CRTORAPKG, 6-8
 - displaying the Oracle SQL Package - panel, 6-7
 - failure due to severity level, 6-9
 - precompile, compile, and bind sample programs, 8-2
 - purpose of command, 6-2
- CRTSQLxxx command
 - Access Manager default user ID, 6-3
 - Calling Oracle Access Manager during package creation, 1-5
 - IBM command for connection management, 8-3
 - IBM command to pre-process file, 6-9
 - known restriction with RDBCNNMTH parameter, 2-4

D

- data areas, blocksize, default block size, 6-4
- Data Manipulation Language (DML), known restriction with two-phase commit, 2-4
- data types
 - conversion of Oracle to DB2/400, 8-6
 - default precision and scale, 2-2
 - LONG and LONG RAW, known restriction, 2-4
 - Oracle data types mapped to DB2/400 data types, 8-6
 - TIME, stored procedure limitations, 2-3
 - TIMESTAMP, stored procedure limitations, 2-3
- DATAQUEUE_TIMEOUT parameter
 - AM400_COM program to dynamically alter AM400_RID program, 7-9
 - description of this tuning parameter, 7-6
 - query the value of, with code example, 7-10
 - tuning parameter caution, 7-6

DATE data type, Oracle data types mapped to DB2/400 data types, 8-6

DB2/400

- commands
 - CONNECT TO, 8-2
 - SET CONNECTION, 8-3
- CONNECT syntax, Oracle CONNECT syntax not supported, 8-3
- conversion of Oracle data types, 8-6
- statements, CONNECT TO, 8-3
- statements, DECLARE PROCEDURE
 - example AS/400 C program that calls Oracle stored procedure GETRESULTS, 8-5
 - must be used when calling an Oracle stored procedure, 8-5

DEAD_CHECK_COUNTER parameter

- description of this tuning parameter, 7-6
- example query, 7-10
- tuning parameter caution, 7-6

debugging

- control string, Debugging Options Panel, 6-7
- options
 - Access Manager Debugging Options Panel (1), 6-6
 - Access Manager Debugging Options Panel (2), 6-7

DECIMAL data type, Oracle data types mapped to DB2/400 data types, 8-6

DECLARE PROCEDURE statement

- DB2/400 statement, stored procedure limitations, 2-3
- example AS/400 C program that calls Oracle stored procedure GETRESULTS, 8-5
- must be used when calling an Oracle stored procedure, 8-5

default

- Access Manager
 - Commands Menu, "Change Oracle Access Manager Default settings" option, 6-2
 - default blocksize, 6-4
 - Default Options Panel, 6-2
 - Default Options Panel, default settings, 6-3
 - default password, 6-4
 - default precision, 6-4
 - default scale, 6-4
 - default user id, 6-3
 - Globalization Support language setting, 6-4
 - activation group, 1-6
 - auxiliary storage pool ID, Install Access Manager Product Panel, 4-3
 - CD drive for installation, 4-2
 - CHGSQLDFLT, changes Access Manager default settings, 6-2
 - data types, precision and scale, 2-2
 - declaring procedures, CALL to a stored procedure, parameters default to IN OUT, 8-5
 - during installation, Access Manager parameter for two-phase commit processing defaults to *NO, 7-1
 - enabling two-phase commit, 7-5
 - for NLS_LANG, B-2
 - installation library name (ORAAM400), 4-2
 - library containing the Access Manager program, 6-3
 - options in Access Manager Network Parameters Panel, 6-5
 - position of Access Manager library name, 4-4
 - precision and scale, Access Manager default number format, 6-4
 - QSECOFR in Privileged ID field, AM400_COM Privileged User IDs, 7-8
 - translating date formats, 1-5
 - user ID and password
 - with STRSQL commands, 8-2
 - values provided during product installation, 4-2
- defining Oracle Access Manager
 - RDB entry, 5-3
 - TNSNAMES entry, 5-3
- defining Oracle server to Access Manager, main topic, 5-2
- DELETE statement, limitation of Oracle hints, 2-3
- DESCRIBE TABLE command, not issued interactively, 8-4
- describing tables, STRSQL restriction, 8-4
- description field, changing AM400_RID options, 7-7
- diagnosing problems, Oracle Access Manager
 - changing debugging settings, 6-6
 - if package creation failed, 6-8
- DISCONNECT command, protected conversation, 8-4
- distributed database environment, 1-5
- distributed unit of work, levels of distributed databases, 1-5
- documentation, recommended, 3-2
- DROP statement, two-phase commit restriction, 6-4
- DSPJOBLOG command, verifying installation, 4-3

E

EDTF command, defining TNSNAMES entry, 5-5

error

- error code processing, Oracle Access Manager, 8-7
- mapped error code, message and error code processing, 8-7
- ORA-0090, invalid CREATE command, 8-2
- ORA-1017, attempting to CONNECT without user ID, 6-4
- ORA-1457, when using STRSQL, 2-2
- ORA-2089, known restriction with two-phase commit, 2-4
- severity level, CRTORAPKG command failure, 6-9
- SQL0105, with CRTORAPKG command, 6-9
- SQL0969, known restriction with two-phase commit processing, 2-3
- SQL5001, with CRTORAPKG command, 6-9
- using SQLCA, message and error code processing, 8-7

eServer, AS/400 and OS/400, platform names for IBM eServer iSeries servers, viii

examples

- illustrations or graphics, see "figures"
- Panel illustrations, see panel name or "panels"
- samples of code entries, see "code example"
- text format conventions, examples of code input and output, ix

EXEC SQL statement, Oracle hints limitation, 2-3

F

features

- main topic, 1-4
- data type conversions, 1-4
- distributed applications, 1-5
- leveraging existing applications, 1-4
- Oracle SQL language, 1-4
- universal access, 1-4

fields

- SQLERRMC, message and error code processing, 8-7
- SQLSTATE, message and error code processing, 8-7

figures

- 1-1, "Oracle Database Gateway", 1-1
- 1-2, "Oracle Access Manager", 1-2
- 1-3, "Oracle Access Manager for AS/400 Architecture", 1-3
- 5-1, "Native IBM Networking", 5-2
- 5-2, "Native Oracle Networking", 5-2
- 5-3, "Oracle Access Manager Networking", 5-2
- 5-4, "Defining Oracle Access Manager to the Oracle Server", 5-3

files

- ORA(TNSNAMES), connecting to an Oracle server, 5-3
- Oracle Access Manager
 - QCBLESRC, installation library that holds sample program file, 8-2
 - QRPGLESRC, installation library that holds sample program file, 8-2
 - QSQLCSRC, installation library that holds sample program file, 8-2

- FLOAT data type, Oracle data types mapped to DB2/400 data types, 8-6
- FOR FETCH ONLY clause, required for block fetching to be performed, 6-4
- FORCE option, Oracle required privileges, 7-9

functions

- precision and scale
 - AVG, 6-4
 - COUNT(*), 6-4
 - MAX, 6-4
 - SUM, 6-4
- TO_CHAR, IBM precompiler flags any reference to non-IBM functions, 6-9
- TO_DATE, IBM precompiler flags any reference to non-IBM functions, 6-9

G

GETRESULTS procedure, example, declaring procedures, 8-5

Globalization Support

- default language setting, 6-4
- languages, table of Oracle NLS_LANG values, B-2

GO CMDORAAM, command to display Access Manager commands menu, 6-1

GRANT command, granting Oracle required privileges, 7-9

H

hardware requirements, 3-1

hints, Oracle hints, general limitations, 2-3

HOLD parameter, known restriction with EXEC SQL COMMIT HOLD command, 2-4

host variable, for Oracle hints (known problem), 2-3

I

IBM

- Application Requester Driver (ARD) interface
 - data from Oracle server is translated by IBM ARD API, B-2
 - Oracle Access Manager architecture, 1-2
- commands
 - CALL, uses of Oracle SQL language, 1-4
 - CRTSQLxxx, connection management, 8-3
- Distributed Relational Databases, 1-5
- eServer iSeries servers, platform name, also AS/400 and OS/400, viii
- IBM ARD, distributed database environment, 1-5
- identify, uniquely identifying a connection, 1-6
- implied commit, defined, 7-2
- IN OUT parameter, stored procedure parameters, 8-5
- IN parameter, stored procedure parameters, 8-5

in-doubt

- automatic processing is controlled by AM400_RID program, 7-9
- caution note regarding parameters that govern resolving in-doubt tasks, 7-6
- changing AM400_RID options, using CHG2PCPRM command, 7-5
- figure 7-1, "AS/400 In-Doubt Transactions", 7-4
- FORCE marks jobs, AM400_COM program command, 7-11
- required privileges for connecting to any Oracle server, 7-8
- transactions
 - main topic, 7-3
 - resolving, 7-3
- use of CLEANUP_DELAY parameter, 7-6
- use of DATAQUEUE_TIMEOUT parameter, 7-6
- use of DEAD_CHECK_COUNTER parameter, 7-6

Install Oracle Access Manager Product - panel, 4-2

installation

- checklist, 4-2
- choosing the library, 4-2
- command to verify successful installation, 4-3
- default values, 4-2
- finishing installation, 4-3
- installation procedure will not install into a library with existing objects, 4-3
- logging on to AS/400, 4-2
- mounting product CD-ROM, 4-2
- selecting installation library name, 4-2
- starting install process, 4-2
- verifying installation parameters, 4-3

INTEGER data type, Oracle data types mapped to DB2/400 data types, 8-6

introduction, Oracle Access Manager, 1-1

invoking the Oracle Access Manager, 1-6

iSeries, AS/400 and OS/400, platform names for IBM eServer iSeries servers, viii

J

job log, checking for error messages after installation of Access Manager, 4-3

K

known problems and restrictions, main topic, 2-2

L

language

- default Globalization Support language setting, 6-4
- table of Oracle NLS_LANG values, B-2
- types, values
 - *C, 6-8
 - *COBOL, 6-8
 - *RPG, 6-8

library

- Oracle Access Manager, adding to library list, 6-1
- the installation procedure will not install into a library with existing objects, 4-3

LIST command, AM400_COM commands, description and example, 7-11

LODRUN command, starting Access Manager installation, 4-2

logical boundary

- for application cleanup processing, provided by an activation group, 1-6
- for message creation, provided by an activation group, 1-6

LONG data type

- known restriction, 2-4
- Oracle data types mapped to DB2/400 data types, 8-6

LONG RAW data type

- known restriction, 2-4
- Oracle data types mapped to DB2/400 data types, 8-6

M

mapped error code, message and error code processing, 8-7

mapping Oracle data types to DB2/400 data types, 8-6

MAX function, precision and scale of NUMBER values, 6-4

members, name format, mapping file names to members, 5-2

messages, Oracle Access Manager, processing, 8-7

MODIFY command, AM400_COM commands, description and example, 7-10

N

network parameters, Access Manager Network Parameters Panel, 6-5

network, requirements, 3-2

NUMBER data type, Oracle data types mapped to DB2/400 data types, 8-6

O

operating system requirements, 3-2

options

- default options in Access Manager Network Parameters Panel, 6-5
- FORCE, Oracle required privileges, 7-9
- Oracle Access Manager
 - USER, if not included in CONNECT TO statement, 6-3
 - USING, if not included in CONNECT TO statement, 6-3

ORA(TNSNAMES) file

- connecting to a remote Oracle server, 5-2
- defining Access Manager to Oracle server, 5-3

ORA-0090 error, invalid CREATE command, 8-2

- ORA-1017 error, attempting to CONNECT without user ID, 6-4
- ORA-1457 error, when using STRSQL, 2-2
- ORA-2089 error, known restriction with two-phase commit, 2-4
- Oracle Access Manager
 - architecture, 1-2
 - back up any existing Access Manager library before upgrading to a new release, 4-2
 - calling, 1-5
 - checklist
 - configuring Oracle Net, TCP/IP AS/400, 5-4
 - enabling two-phase commit, 7-4
 - installation, 4-2
 - post-installation, 4-4
 - commands
 - main topic, 6-1
 - ADDLIB, 6-1
 - CALL, 8-2, 8-4
 - CHG2PCPRM, 6-9
 - CHGSQLDBG, 6-6
 - CHGSQLDFLT, 6-2, 8-2, B-3
 - COMMIT, 6-5
 - CRTORAPKG, 6-7, 8-2
 - CRSQLxxx, 6-3
 - DESCRIBE TABLE, 8-4
 - DISCONNECT, 8-4
 - ROLLBACK, 6-5
 - SELECT, 8-4
 - Commands Menu, 6-2
 - commitment control resource, 6-5
 - configuring, 6-1
 - connecting
 - connect processing, 1-5
 - connection management, 8-3
 - from AS/400 precompiled application, 8-2
 - to Oracle server, 8-1
 - conversion of Oracle data types to DB2/400 data types, 8-6
 - DB2/400
 - CONNECT syntax, Oracle CONNECT syntax not supported, 8-3
 - statements, CONNECT TO, 8-3
 - debugging, 6-6
 - debugging control string, Debugging Options Panel, 6-7
 - Debugging Options, 6-6
 - defining Oracle server to Access Manager, 5-2
 - describing tables, 8-4
 - diagnosing problems
 - debugging options panel, 6-6
 - if package creation failed, 6-8
 - error code processing, 8-7
 - explicit connection, 8-3
 - features, 1-4
 - files
 - QCBLESRC, 8-2
 - QRPGLESRC, 8-2
 - QSQLCSRC, 8-2
 - Globalization Support support, B-1
 - GO CMDORAAM displays commands menu, 6-2
 - implicit connection, 8-3
 - installation library
 - add to your library list, 4-4
 - default position is *FIRST, 6-1
 - introduction, 1-1
 - library
 - adding to list, 6-1
 - name, default position, 4-4
 - message processing, 8-7
 - options
 - USER, if not included in CONNECT TO statement, 6-3
 - USING, if not included in CONNECT TO statement, 6-3
 - package creation, 1-5
 - parameters
 - RDB, 8-3
 - stored procedures, 8-5
 - stored procedures, IN, 8-5
 - stored procedures, IN OUT, 8-5
 - stored procedures, OUT, 8-5
 - performance, SQL*Net trace, 6-5
 - programs
 - SAMP1C, 8-2
 - SAMP1CBL, 8-2
 - SAMP1RPG, 8-2
 - sample, C, A-1
 - sample, COBOL, A-3
 - sample, RPG, A-4
 - RELEASE ALL command, 8-3
 - SQL statement processing, 1-5
 - statements
 - CALL, advantages of stored procedures, 8-4
 - CALL, executed interactively or embedded, 8-4
 - COMMIT, connection management, coordinating transactions, 8-3
 - COMMIT, connection management, ending all connections, 8-3
 - COMMIT, with multiple connections open, 8-3
 - CONNECT TO, Access Manager default user id, 6-3
 - DDL, CREATE, two-phase commit, 6-4
 - DDL, DROP, two-phase commit, 6-4
 - ROLLBACK, connection management, 8-3
 - ROLLBACK, connection management note, 8-4
 - ROLLBACK, with multiple connections open, 8-3
 - SELECT, performing block fetching, 6-4
 - SELECT, verifying that rows were inserted successfully, 8-2
 - stored procedures
 - main topic, 8-4
 - advantages, 8-4
 - declaring, 8-5

- GETRESULTS, 8-5
- table of component interactions, 1-3
- trace file, Debugging Options Panel, 6-7
- trace library, Debugging Options Panel, 6-7
- tracing values
 - *ADMIN, 6-6
 - *OFF, 6-6
 - *USER, 6-6
 - 16, 6-6
- two-phase commit, default options panel, 6-4
- Oracle CONNECT syntax is not supported by Oracle Access Manager, 8-3
- Oracle hints, general limitations, 2-3
- Oracle Net
 - Access Manager component interactions, 1-4
 - checklists, TCP/IP AS/400 configuration, 5-4
 - commands
 - ADDRDBDIRE, adding an entry to the RDB directory, 5-6
 - ADDRDBDIRE, defining IBM DRDA connection, 5-2
 - WRKMBRPDM, 5-5
 - configuration, TCP/IP, 5-4
 - connecting through Oracle Net, 5-2
 - file names, 5-1
 - parameters, TNSNAMES.ORA, 5-1
 - post-installation configuration, 4-5
 - trace options, 6-5
- Oracle server
 - Access Manager component interactions, 1-3
 - connecting to, 8-1
 - defining Oracle server to Access Manager, 5-2
 - Oracle Server Name field, 7-7
- Oracle SQL Package Panel, 6-8
- OS/400 and AS/400, platform names for IBM eServer iSeries servers, viii
- OUT parameter, stored procedure parameters, 8-5

P

- package creation
 - checking problems if package creation fails, 6-8
 - with Oracle Access Manager, 1-5

- panels
 - Access Manager Commands Menu, 6-2
 - Access Manager Debugging Options Panel (1), 6-6
 - Access Manager Debugging Options Panel (2), 6-7
 - Access Manager Default Options Panel, 6-2, 6-3
 - Access Manager Network Parameters Panel, 6-5
 - ADDRDBDIRE Panel_1, 5-4
 - ADDRDBDIRE Panel_2, 5-5
 - Change or Enter New Server Attributes, 7-6
 - Change Privileged IDs (Relative to AM400_COM), 7-8
 - Change RID (Resolve In Doubt) Options, 7-5
 - Change RID (Resolve In Doubt) Options Panel, 6-9
 - default values provided during product installation, 4-2
 - Install Oracle Access Manager Product, 4-2
 - Oracle SQL Package Panel, 6-8
 - Verify Oracle Access Manager Install, 4-3
- parameters
 - CLEANUP_DELAY
 - caution, changing the AM400_RID options, 7-6
 - changed by AM400_COM program, 7-9
 - querying the set value, 7-10
 - tuning parameter description, 7-6
 - DATAQUEUE_TIMEOUT
 - caution, changing the AM400_RID options, 7-6
 - changed by AM400_COM program, 7-9
 - querying the set value, 7-10
 - tuning parameter description, 7-6
 - DEAD_CHECK_COUNTER
 - caution, changing the AM400_RID options, 7-6
 - querying the set value, 7-10
 - tuning parameter description, 7-6
 - HOLD, known restriction with EXEC SQL COMMIT HOLD command, 2-4
 - Oracle Access Manager
 - PASSWORD, default options, 6-3
 - RDB, connection management, 8-3
 - USER, default options, 6-3
 - ROBCNNMTH, known restriction, 2-4
 - SERVER_STARTUP_TIMEOUT, querying for resolve-in-doubt transaction, 7-11
 - SQL*Net, TNSNAMES.ORA, 5-1
 - STARTUP
 - description, 7-7
 - uses of query, AM400_COM commands, 7-10
 - stored procedures
 - main topic, 8-5
 - IN, 8-5
 - IN OUT, 8-5
 - OUT, 8-5
 - test connectivity before changing any configuration parameters, 4-5
 - TRACE_LEVEL

- determines trace level for AM400_RID
 - program, 7-6
 - querying the set value, 7-11
 - verify installation parameters, 4-3
- password
 - default user ID and password
 - with STRSQL commands, 8-2
 - for Oracle IDs field, Change or Enter New Server Attributes Panel, 7-7
- phrases
 - SIMPLE CALL WITH NULLS, stored procedure limitations, 2-3
 - SIMPLE CALL, stored procedure limitations, 2-3
- platform, iSeries, AS/400 and OS/400, platform names for IBM eServer iSeries servers, viii
- port number of the Oracle Net listener, usually 1521, 5-6
- position, default position of Access Manager library name, 4-4
- post-installation
 - main topic, 4-4
 - add Oracle Access install library to library list, 4-4
 - checklist, 4-4
 - configure SQL*Net, 4-5
 - configure two-phase commit (optional), 4-5
 - retest connectivity to Oracle server, 4-5
 - test connectivity to Oracle server, 4-5
- precision
 - default data type, 2-2
 - default precision and scale, Access Manager default number format, 6-4
- prepare phase, defined, two-phase commit processing terminology, 7-2
- PREPARE, with a COMMIT or ROLLBACK command, known restriction, 2-3
- privileged Oracle IDs field, Change or Enter New Server Attributes Panel, 7-7
- privileges
 - GRANT command, 7-9
 - required by Oracle, 7-8
- procedures
 - CALL, declaring procedures, parameter defaults, 8-5
 - GETRESULTS, example, extract from an AS/400 C program, 8-5
 - installation, the installation procedure will not install into a library with existing objects, 4-3
- processor, requirements, 3-1
- product set, 2-1
- program
 - SAMP1C program, 8-2
 - SAMP1CBL program, 8-2
 - SAMP1RPG program, 8-2
 - sample programs
 - Oracle Access Manager, C, 8-2, A-1
 - Oracle Access Manager, COBOL, 8-2, A-3
 - Oracle Access Manager, RPG, 8-2, A-4
- programs
 - AM400_COM

- main topic, 7-9
- dynamically add or remove Oracle server parameters, 7-7
- AM400_RID
 - listing servers and connections, 7-11
 - resolving in-doubt transactions, 7-3
- Oracle Access Manager
 - SAMP1C, 8-2
 - SAMP1CBL, 8-2
 - SAMP1RPG, 8-2
- sample
 - C, A-1
 - COBOL, A-3
 - RPG, A-4

Q

- QCBLESRC file, sample program files, 8-2
- QRPGLESRC file, sample program files, 8-2
- QSECOFR
 - security administration user ID, 7-8
 - user profile, authority for SECADM (before installing Oracle Access Manager), 4-2
- QSQLCSRC file, sample program files, 8-2
- QTEMP, Access Manager trace library value, 6-7
- QUERY command, queries various items within AM400_RID environment, examples provided, 7-10

R

- RAW data type, Oracle data types mapped to DB2/400 data types, 8-6
- RDB
 - directory
 - add entry with ADDRDBDIRE command, 5-6
 - defining TNSNAMES entry, 5-6
 - entry, remote database directory entry, 5-3
 - name, connections and activation groups, 1-6
 - parameter, connection management, 8-3
 - recommendation
 - back up any existing Access Manager library before upgrading to a new release, 4-2
 - keep up to date with the latest DB2/400 PTF group, 3-2
 - use the default password only for testing, 6-4
- RELEASE ALL command, connection management, 8-3
- RELEASE command, connection management, 8-3
- remote unit of work, levels of distributed databases, 1-5
- REMOVE command, description, AM400_COM commands, 7-10
- requirements
 - communications, 3-2
 - hardware
 - CD-ROM drive, 3-1
 - disk space, 3-1
 - main topic, 3-1

- processor, 3-1
- software
 - additional, 3-2
 - communications, 3-2
 - main topic, 3-2
 - operating system, 3-2
 - Oracle server, 3-2
 - PTF, 3-2
- RID
 - Change RID (Resolve In Doubt) Options Panel, 6-9
- ROBCNNMTH parameter, known restriction, 2-4
- ROLLBACK command
 - Access Manager default options, commitment control boundary, 6-5
 - ending a unit of work (sequence of SQL commands), 1-6
 - known restriction for two-phase commit processing, 2-3
 - Oracle required privileges, with FORCE option, 7-9
 - two-phase commit processing, 7-3
- ROLLBACK statement
 - connection management, 8-3
 - with DB2/400 RELEASE command (a note), 8-4
- ROWID data type, Oracle data types mapped to DB2/400 data types, 8-6
- runtime data structures, provided by an activation group, 1-6

S

- SAMP1C program, 8-2
- SAMP1CBL program, 8-2
- SAMP1RPG program, 8-2
- sample programs
 - Oracle Access Manager
 - C, A-1
 - COBOL, A-3
 - RPG, A-4
- scale
 - default data type, 2-2
 - default precision and scale, Access Manager default number format, 6-4
- SELECT statement
 - describing tables (STRSQL), 8-4
 - limitation of Oracle hints, 2-3
 - performing block fetching, 6-4
 - verifying that rows were inserted successfully, 8-2
- SERVER_STARTUP_TIMEOUT parameter, 7-11
- SET command, AM400_COM commands, description and example, 7-11
- SET CONNECTION command, connection management, 8-3
- severity level, CRTORAPKG command failure, 6-9
- SHUTDOWN command, description, AM400_COM command, 7-10
- SIMPLE CALL phrase is supported, stored procedure limitations, 2-3

- SIMPLE CALL WITH NULLS phrase not supported, stored procedure limitations, 2-3
- SMALLINT data type, Oracle data types mapped to DB2/400 data types, 8-6
- software requirements, 3-2
- SQL
 - all SQL connections are managed at the activation group level., 1-6
 - client manager, Access Manager component interactions, 1-3
 - Development Kit, required in order to run STRSQL, 3-2
 - Oracle SQL Package Panel, 6-8
 - SQL0105 error, with CRTORAPKG command, 6-9
 - SQL0969 error, known restriction with two-phase commit, 2-3
 - SQL5001 error, with CRTORAPKG command, 6-9
 - SQLCA, message and error code processing, 8-7
 - SQLERRMC field, message and error code processing, 8-7
 - SQLSTATE field, message and error code processing, 8-7
 - statements, processing on Oracle Access Manager, 1-5
- STARTUP command, description, AM400_COM command, 7-10
- STARTUP parameter
 - enabling two-phase commit processing, 7-7
 - queries, AM400_COM program, 7-10
- startup parameter field, enabling two-phase commit processing, 7-7
- statements
 - CONNECT, known restriction with two-phase commit, 2-4
 - DB2/400
 - CONNECT TO, 8-3
 - DECLARE PROCEDURE, 8-5
 - DECLARE PROCEDURE, DB2/400 statement, stored procedure limitations, 2-3
 - DELETE, 2-3
 - EXEC SQL, Oracle hints limitation, 2-3
 - Oracle Access Manager
 - CALL, 8-4
 - COMMIT, 8-3
 - CONNECT TO, 6-3
 - DDL, CREATE, 6-4
 - DDL, DROP, 6-4
 - ROLLBACK, 8-3, 8-4
 - SELECT, 8-2
 - SELECT, performing block fetching, 6-4
 - SELECT, 2-3
 - SQL, processing, 1-5
 - UPDATE, 2-3
- STATUS command, requests global connection status of AM400_RID program, 7-9
- status signal, transmitted for every connect and disconnect, and timeout, 7-6

- stored procedures
 - main topic, 8-4
 - advantages, 8-4
 - declaring procedures, 8-5
 - GETRESULTS procedure, 8-5
 - limitations, 2-3
 - parameters
 - main topic, 8-5
 - IN, 8-5
 - IN OUT, 8-5
 - OUT, 8-5
 - STRSBS command, enabling two-phase commit
 - processing, 7-8
 - STRSQL
 - activation group, calling Access Manager, 1-6
 - checking connection status, 6-8
 - connecting to an Oracle server, 8-1
 - SUM function, precision and scale, 6-4
 - SYS.PENDING_SESSIONS\$ table, 7-9
 - SYS.PENDING_TRANS\$ table, 7-8
 - system requirements
 - hardware, 3-1
 - processor, 3-1
 - software
 - additional software, 3-2
 - DB2/400 Query Manager, 3-2
 - network, 3-2
 - operating system, 3-2
-
- T**
- tables
 - 1-1, Access Manager Component Interactions, 1-3
 - 6-1, Access Manager Commands, 6-2
 - 8-1, Sample Program Files, 8-2
 - 8-2, Oracle data types mapped to DB2/400 data types, 8-6
 - 8-3, Error Code Mapping, 8-7
 - B-1, Oracle Character Set Names, B-1
 - B-2, Oracle NLS_LANG Values, B-2
 - SYS.PENDING_SESSIONS\$, 7-9
 - SYS.PENDING_TRANS\$, 7-8
 - TCP/IP
 - AS/400 configuration
 - define TNSNAMES entry, 5-5
 - configuring SQL*Net, 5-4
 - connecting the AS/400 to an Oracle server, 1-2
 - terminology, two-phase commit
 - AM400_COM program, 7-1
 - AM400_RID program, 7-1
 - commit phase, 7-2
 - commit point site, 7-2
 - implied commit, 7-2
 - prepare phase, 7-2
 - two-phase commit parameter, 7-2
 - test connectivity before changing any configuration parameters, 4-5
 - text format conventions for code input and output, ix
 - TIME data type, stored procedure limitations, 2-3
 - timeout, DATAQUEUE_TIMEOUT parameter,
 - description, with AM400_RID program and status signal, 7-6
 - TIMESTAMP data type
 - Oracle data types mapped to DB2/400 data types, 8-6
 - stored procedure limitations, 2-3
 - TNS (Transparent Networking Substrate), with Oracle Net, 5-1
 - TNSNAMES
 - defining for TCP/IP, 5-5
 - entry, 5-3
 - TNSNAMES.ORA parameter, 5-1
 - TO_CHAR function, 6-9
 - TO_DATE function, 6-9
 - tracing
 - trace file, Debugging Options Panel, 6-7
 - trace library, Debugging Options Panel, 6-7
 - TRACE_LEVEL parameter
 - on Change RID (Resolve In Doubt) Options Panel, 7-6
 - with QUERY, 7-11
 - values
 - *ADMIN, 6-6
 - *OFF, 6-6
 - *USER, 6-6
 - 16, 6-6
 - translating date formats, default for IBM AS/400, 1-4
 - Transparent Networking Substrate (TNS), with Oracle Net, 5-1
 - tuning parameter, DATAQUEUE_TIMEOUT,
 - description, with AM400_RID program, 7-6
 - two-phase commit
 - AM400_COM program
 - adding and removing privileged user IDs, 7-7
 - communicating through a shared data space, 7-11
 - main topic, 7-9
 - user ID in the Privileged ID field, 7-8
 - AM400_RID program
 - altering environment with AM400_COM program, 7-9
 - commit phase, announcing the end of a transaction, 7-3
 - communication with AM400_COM program, 7-9
 - connecting to a designated server, 7-7
 - connections with servers, 7-11
 - designate connections to Oracle servers (AM400_COM function), 7-7
 - prepare phase, 7-3
 - runs in its own subsystem, 7-3
 - starting the AM400_RID program subsystem with STRSBS, 7-8
 - status signal and timeout, 7-6
 - commands
 - AM400_COM program, main topic, 7-9
 - AM400_COM program, ADD, 7-10
 - AM400_COM program, FORCE, 7-11

- AM400_COM program, LIST, 7-11
 - AM400_COM program, MODIFY, 7-10
 - AM400_COM program, QUERY, 7-10
 - AM400_COM program, REMOVE, 7-10
 - AM400_COM program, SET, 7-11
 - AM400_COM program, SHUTDOWN, 7-10
 - AM400_COM program, STARTUP, 7-10
 - AM400_COM program, STATUS, 7-9
 - CHG2PCPRM, 7-5, 7-7
 - CHGSQLDFLT, 7-1
 - COMMIT, phase 2, commit or abort and roll back, 7-3
 - COMMIT, prepare phase, 7-2
 - COMMIT, privileges, 7-9
 - GRANT, 7-9
 - ROLLBACK, phase 2, commit or abort and roll back, 7-3
 - ROLLBACK, privileges, 7-9
 - STRSBS, 7-8
 - default options panel, 6-4
 - enabling
 - main topic, 7-4
 - add or remove AM400_COM privileged user IDs, 7-7
 - change Oracle Access Manager commit default, 7-5
 - change the AM400_RID options, 7-5
 - checklist, 7-4
 - start the AM400_COM program (optional), 7-8
 - start the subsystem that runs AM400_RID program, 7-8
 - fields
 - description, 7-7
 - Oracle server name, 7-7
 - password for Oracle ID, 7-7
 - privileged Oracle ID, 7-7
 - startup parameter, 7-7
 - overview, 7-1
 - panels
 - Change or Enter New Server Attributes, 7-6
 - change privileged IDs (Relative to AM400_COM), 7-8
 - change RID (Resolve In Doubt) Options, 7-5
 - parameters
 - CLEANUP_DELAY, 7-6, 7-9, 7-10
 - DATAQUEUE_TIMEOUT, 7-6, 7-9, 7-10
 - DEAD_CHECK_COUNTER, 7-6, 7-10
 - parameter, defined, 7-2
 - SERVER_STARTUP_TIMEOUT, 7-11
 - STARTUP, 7-7, 7-10
 - TRACE_LEVEL, 7-6, 7-11
 - post-installation, configure two-phase commit (optional), 4-5
 - process
 - main topic, 7-2
 - commit phase, 7-3
 - phase 1, 7-3
 - phase 2, 7-3
 - prepare phase, 7-3
 - resolve in-doubt transactions, 7-3
 - restrictions with SQL, 1-4
 - tables
 - SYS.PENDING_SESSIONS\$, 7-9
 - SYS.PENDING_TRANS\$, 7-9
 - terminology
 - main topic, 7-1
 - AM400_COM program, 7-1
 - AM400_RID program, 7-1
 - commit phase, 7-2
 - commit point site, 7-2
 - implied commit, 7-2
 - prepare phase, 7-2
 - two-phase commit parameter, 7-2
- ## U
-
- unit of work, defined, 1-6
 - UPDATE statement, limitation of Oracle hints, 2-3
 - user ID, default user ID and password
 - with STRSQL commands, 8-2
 - USER option, if not included in CONNECT TO statement, 6-3
 - USER parameter, default options, 6-3
- ## V
-
- VARCHAR data type, Oracle data types mapped to DB2/400 data types, 8-6
 - VARCHAR2 data type, Oracle data types mapped to DB2/400 data types, 8-6
 - Verify Oracle Access Manager Install - panel, 4-3
- ## W
-
- WRKMBRPDM command, command to edit member TNSNAMES in ORA file, with example code, 5-5

