

Oracle® Migration Workbench

Frequently Asked Questions (FAQ)

Release 9.2.0 for Microsoft Windows 98/2000 and Microsoft Windows NT

March 2002

Part No. A97247-01

ORACLE®

Part No. A97247-01

Copyright © 1996, 2002 Oracle Corporation. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information of Oracle Corporation; 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. Oracle Corporation does not warrant that this document is 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, without the express written permission of Oracle Corporation.

If the Programs are delivered to the U.S. Government or anyone licensing or using the programs on behalf of the U.S. Government, the following notice is applicable:

Restricted Rights Notice Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the Programs, including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication, and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 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 Oracle Corporation disclaims liability for any damages caused by such use of the Programs.

Oracle is a registered trademark, and Oracle8, Oracle8i, Oracle9i, SQL*Plus, PL/SQL, and Pro*C are trademarks or registered trademarks of Oracle Corporation. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments	v
Preface.....	vii
1 Oracle Migration Workbench Frequently Asked Questions	
Overview	1-2
Pre-Installation	1-2
Installation and Configuration	1-5
Data Migration	1-7
Support	1-15
2 Informix Dynamic Server Questions	
Installation and Configuration	2-2
Data Migration	2-3
3 Microsoft Access Questions	
Pre-Installation	3-2
Installation and Configuration	3-2
Data Migration	3-4
4 Microsoft SQL Server and Sybase Adaptive Server Questions	
Pre-Installation	4-2
Data Migration	4-5

Data Types	4-8
Parsing	4-8
Global Variables	4-21
Triggers and Rules	4-24

5 MySQL Server Questions

Data Migration	5-2
Migration Error Messages	5-2

A List of Questions

General Migration Workbench Questions.....	A-1
Informix Dynamic Server Questions	A-3
Microsoft Access Questions	A-5
Microsoft SQL Server & Sybase Adaptive Server Questions	A-6
MySQL Questions	A-8

Index

Send Us Your Comments

Oracle Migration Workbench Frequently Asked Questions (FAQ), Release 9.2.0 for Microsoft Windows 98/2000 and Microsoft Windows NT

Part No. A97247-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Email - infomwb_ww@oracle.com
 - Tel: +353-1-8031000, Fax: +353-1- 8033321
 - Attn: Oracle Migration Workbench
- Postal service:
Oracle Migration Workbench Documentation
Oracle Corporation
Block P5
East Point Business Park
Clontarf, Dublin 3
Ireland

Preface

The Oracle Migration Workbench Frequently Asked Questions (FAQ) provides questions and answers about to the Oracle Migration Workbench. It is a useful guide regardless of the conversion tool you are using to perform a database migration, however Oracle Corporation recommends that you use the Migration Workbench to migrate from your existing database platform to an Oracle Server. This preface contains the following sections:

- [Audience](#)
- [What You Should Already Know](#)
- [Documentation Accessibility](#)
- [Conventions Used in This Guide](#)

Audience

This guide is intended for anyone migrating a Microsoft Access, Informix Dynamic Server, Microsoft SQL Server, Sybase Adaptive Server, or MYSQL Server database to Oracle8i or higher.

What You Should Already Know

You should be familiar with relational database concepts and with the operating system environment of your Oracle Server and the database that you want to migrate.

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. Standards will continue to evolve over time, and Oracle Corporation 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 additional information, visit the Oracle Accessibility Program Web site at

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

Accessibility of Code Examples in Documentation

JAWS, a Windows screen reader, 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, JAWS may not always read a line of text that consists solely of a bracket or brace.

Conventions Used in This Guide

The following typographic conventions are used in this guide:

Convention	Description
UPPERCASE	Uppercase text indicates case-insensitive file names or directory names, commands, command keywords, initialization parameters, data types, table names, or object names. Enter text exactly as spelled.
UPPERCASE in Code Examples	Uppercase words in example statements indicate the keywords within Oracle SQL. When you issue statements, however, keywords are not case sensitive.
lowercase in Code Examples	Lowercase words in example statements indicate words supplied only for the context of the example. For example, lowercase words may indicate the name of a table, column, or file.
[]	In text, key names are represented by uppercase letters enclosed in brackets, for example [RETURN]. In command syntax, brackets indicate optional items: <code>cvtrct <i>termname</i> [<i>outfile</i>]</code>
<i>Italics</i>	Italic type indicates variables, including variable portions of file names. It is also used for emphasis.
Code Examples	Monospace type indicates commands, directory names, user names, path names, file names, and text quoted from screen output. Monospace text must be entered exactly as shown. Example statements may include punctuation, such as commas or quotation marks. All punctuation in example statements is required. All example statements terminate with a semicolon (;). Depending on the application, a semicolon or other terminator may or may not be required to end a statement.
Bold	Boldface type indicates emphasis.
{ }	Braces indicate required items: <code>.DEFINE {macro1}.</code> Do not enter the braces.
	A vertical line indicates a choice within braces or brackets: <code>SIZE filesize [K M]</code> Do not enter the vertical bar.

Convention	Description
...	Ellipses indicate an arbitrary number of similar items: <code>CHKVAL fieldname value1 value2 ... valueN</code> Do not type the ellipsis.
SQL*Plus prompt	The SQL*Plus prompt, <code>SQL></code> , appears in SQL statements and SQL*Plus command examples. Do not enter the prompt when entering the examples.
Microsoft-DOS prompt	The Microsoft-DOS prompt, <code>></code> , appears in Microsoft-DOS command examples. Do not enter the prompt when entering the examples.
Storage measurements	Storage measurements use the following abbreviations: K, for kilobyte which equals 1024 bytes M, for megabyte which equals 1 048 576 bytes G, for gigabyte which equals 1 073 741 824 bytes

Oracle Migration Workbench Frequently Asked Questions

This chapter contains frequently asked questions about using the Oracle Migration Workbench release 9.2.0 for Microsoft Windows 98/2000 and Microsoft Windows NT. It contains the following sections:

- [Overview](#)
- [Installation and Configuration](#)
- [Data Migration](#)
- [Support](#)

See Also: For database platform specific questions, see the following chapters:

- [Chapter 2, "Informix Dynamic Server Questions"](#)
- [Chapter 3, "Microsoft Access Questions"](#)
- [Chapter 4, "Microsoft SQL Server and Sybase Adaptive Server Questions"](#)
- [Chapter 5, "MySQL Server Questions"](#)

Overview

Use the Oracle Migration Workbench release 9.2.0 for Microsoft Windows 98/2000 and Microsoft Windows NT to migrate to Oracle9i or Oracle8i, from the following database platforms:

- Informix Dynamic Server 7.3
- Microsoft Access 2.0, 95, 97, 2000
- Microsoft SQL Server 6.5, 7.0, 2000
- Sybase Adaptive Server 11, 12
- MySQL 3.22, 3.23

The questions in this chapter apply to all of these database platforms.

Pre-Installation

This section contains Migration Workbench pre-installation questions.

What release of the Oracle Server does the Migration Workbench support?

The Migration Workbench supports the following Oracle Server releases:

- Oracle8i release 3 (8.1.7)
- Oracle9i release 1 (9.0.1)

What platforms does Oracle Migration Workbench run on?

The Migration Workbench runs on the following platforms:

- Windows NT 4.0 (Service Pack 5 minimum)
- Windows 98
- Windows 2000

Does the Migration Workbench support UNIX?

The Migration Workbench runs on Windows NT and Windows 98/2000. However, you can use the Migration Workbench to migrate from a supported source database to an Oracle9i or Oracle8i database, regardless of the operating system platform of either database. To do this successfully, you must have a JDBC/ODBC connection to the source database and a SQL*Net connection to the destination Oracle database. However, if the Migration Workbench is remote to the source and des-

tionation databases, you should use the offline data loading capability when you migrate the data.

Can I download the Migration Workbench from the Web?

Yes. You can download the latest release of the Migration Workbench, patches, upgrades, support documentation, and this FAQ from the OTN Web site at:

<http://otn.oracle.com/tech/migration/workbench>

Why am I not able to access Oracle Technology Network (OTN)?

The firewall that the company you work for might prevent you from accessing Oracle Technology Network (OTN). Contact the system administrator to see if there is a restriction that prevents you from downloading files outside of the company firewall.

What should I do if I cannot download the Migration Workbench from Oracle Technology Network (OTN)?

If the maximum number of connections available on the Oracle Server is reached, you cannot download the Migration Workbench from the Oracle Technology Network (OTN) Web site. Keep reloading the page until the Save As dialog box appears, then save the Migration Workbench.

If the Save As dialog box does not appear, send an email message to the Migration Workbench support team at infomwb_ww@oracle.com.

Why do I receive a blank DOS window or one of the following error messages while launching the Migration Workbench from Windows 98?

- Out of environment space
- Bad command or file name

These messages appear if the Migration Workbench installation is incomplete, if the DOS window Command Prompt Properties memory setting is too low, or if there are limitations in Windows 98 that prevent the Migration Workbench `omwb.bat` BAT file from successfully executing. To correct these problems:

- Change the Command Prompt Properties memory setting:
 1. Choose **Start > Programs > Command Prompt**.
 2. Right-click on the title bar of the DOS window and choose **Properties...**
 3. Select the Memory tab.

4. Change the Initial environment property from Auto to 1024.
 5. Click **Apply**.
 6. Click **OK** to accept the changes to the memory settings.
 7. Click **OK**.
 8. Start the Migration Workbench.
- If you changed the DOS window Command Prompt memory setting and you still experience problems launching the Migration Workbench, do the following:
 1. Copy the %ORACLE_HOME%/bin/omwb.bat file to %ORACLE_HOME%/bin/omwb_copy.bat. Use this copy if you need to restore the original file.
 2. In the %ORACLE_HOME%/bin/omwb.bat file, locate the path for your source database.

Towards the bottom of the file there is a line beginning with %JRE% -classpath.

Below this line, there is a corresponding path for each plug-in. The following is the path for SQL Server 6:

```
D:\ora816\0mwb\plugins\SQLServer6.jar;
```
 3. Delete the paths for all plug-ins except the one that corresponds to the database you are migrating.
 4. Save the omwb.bat file.

Where can I find out more about Oracle Migration Workbench error messages?

For information on error messages, refer to the Troubleshooting section of the *Oracle Migration Workbench Online Help*.

What is the difference between Oracle Transparent Gateway (OTG) and the Migration Workbench?

Use the Migration Workbench for one-time migrations to an Oracle Server because it migrates all table data. You cannot migrate only the differences between several migrations.

Use the Oracle Transparent Gateway if you want to have links to other heterogeneous databases.

Does the Migration Workbench provide NLS support?

The Migration Workbench has not been tested for NLS support. However, the Migration Workbench has successfully migrated databases with non-English language characters. If you use the Migration Workbench to migrate data or schema object names that contain non-English language characters, you should verify that all data has migrated correctly.

Does the Migration Workbench support object names with spaces in their names?

Yes. If you have an object with a space in the name, convert the space to an underscore (_) by putting it in square brackets []. The Migration Workbench can then migrate the object.

Does the Migration Workbench use strong REF CURSORS?

The Migration Workbench uses weak REF CURSORS. You can alter these to use packages and strong REF CURSORS. To improve the readability of the migrated application, release 9.2.0 of the Migration Workbench supports Weak Ref Cursors.

Installation and Configuration

This section contains Migration Workbench installation and configuration questions.

Can I install the Migration Workbench into an Oracle home directory that contains a release of the Oracle Server earlier than Oracle9i?

No. The Migration Workbench contains Oracle9i release 1 dependencies. Therefore, you can only install the Migration Workbench into an Oracle9i release 1 home directory or into a new Oracle home directory. If you install the Migration Workbench into a new Oracle home directory, and there is already an Oracle server installed on the system you must launch the Home Selector from the Oracle Installation Products program group, to switch the default Oracle home directory to specify the previous Oracle installation. You must do this so that your system does not pick up the new client dependencies for other applications that you might use on the system.

If you attempt to install the Migration Workbench into an Oracle home directory that contains an Oracle Server release earlier than Oracle9i, the Oracle Universal Installer displays the following error message:

The Migration Workbench can only be installed into a new Oracle home directory or an existing Oracle9i Oracle home directory.

How do I create a new Oracle Migration Workbench repository?

You can create the Oracle Migration Workbench repository in an Oracle9i, Oracle8i, or Oracle8 server. To create a new Oracle Migration Workbench repository:

1. Use Oracle DBA Studio, Oracle Enterprise Manager, or Oracle SQL*Plus to create a user in the destination Oracle database where you want to create the Oracle Migration Workbench Repository.

The first time you log in to this user account, you are prompted to create the Oracle Migration Workbench repository.

2. In the Oracle Migration Workbench Repository Login dialog box, enter the user name, password, and service where you want to create the repository.

What should I do if the Migration Workbench Repository creation fails?

If the Migration Workbench Repository fails, you might receive one of the following error messages:

- The Migration Workbench was unable to successfully create the repository. Check the Oracle alertORACLE_SID.log for errors. Failed to create a Migration Workbench repository.
- The Migration Workbench was unable to successfully create the repository. Check the Oracle alertORACLE_SID.log for errors.

The Migration Workbench creates an incomplete repository when it is unable to create all of the required tables. To fix an incomplete repository, recreate a new Migration Workbench repository.

How do I re-create an Oracle Migration Workbench Repository?

You must drop the user in the destination Oracle database, then re-create the user in the destination Oracle database where you want to store the repository. The first time you log in to the newly created user account, you are prompted to create the repository.

How do I create a tnsnames.ora file?

After you install the Migration Workbench, but before you run it, you must set up the `tnsnames.ora` configuration file. You can do this manually or through the Net8 Configuration Assistant. Launch the Net8 Configuration Assistant, then follow the instructions in the wizard and the online help.

What should I do if I cannot launch the Oracle Migration Workbench? I

attempt to do so but nothing happens.

This may occur because the Oracle Java Runtime Environment has been installed into a non-default location. The default location is specified in the `omwb.bat` file in the Oracle Migration Workbench Oracle home. To resolve this issue open the `omwb.bat` file in a text editor and change the `jre` directory path to point to the correct `jre` file location.

I attempt to launch the Oracle Migration Workbench and the splash screen appears then quickly disappears. Why does this happen?

This may occur because there are two instances of the same Oracle home directory in the `tnsnames.ora` file. To resolve this issue open the `%ORACLE_HOME%/omwb/network/admin/tnsnames.ora` file, and remove the second instance of the Oracle home

Data Migration

This section contains Migration Workbench data migration questions.

Can I migrate single table data?

Yes. You can select a table in the Migration Workbench and migrate the data for that table only. To migrate single table data, select **Object > Migrate Table Data**.

Note: The Migrate Table Data option is enabled after you have used the Migration Wizard to create the users and tables in the destination Oracle database.

Can I change the data type of a single column?

When you have created the Oracle Model, you can modify column types from the individual table you created or from all tables. Do this from the General tab of the Property sheet.

Are constraints disabled when loading the data?

Yes. By default, the Migration Wizard first creates the users and their tables, then loads the data, and finally creates all constraints. This avoids problems. It is possible to migrate the database to an Oracle database in stages. For example, you can first create the users and their tables, then create the constraints, and finally load the data. However, doing this might cause problems as a result of referential integrity. Therefore, Oracle Corporation recommends the default implementation.

How are duplicate object names handled?

Duplicate object names are appended with an underscore (_) followed by a number.

What are the object naming guidelines for the Migration Workbench?

Use the following object naming guidelines when you are naming databases or tablespaces:

- Use upper-case characters.
- Begin the name with an alphabetic character.
- Use only alphanumeric characters from the database character set and an underscore (_).
- Do not include quotation marks in the name.

How does the Migration Workbench handle reserved words?

If a column name is an Oracle Server reserved word, the Migration Workbench appends an underscore (_) when you create the object within an Oracle database. You can use ANSI names for objects by choosing **Tools>Options**, then selecting **Create ANSI-compliant names** from the General page. This means that all object names are surrounded by double quotes when you use the Migration Wizard to create them in the destination Oracle database.

If you are using words that are not official T-SQL reserved words, but that have meaning in T-SQL syntax, you must enable the parser to recognize these words. Select the **Allow 'Reserved Words' in table names** option from the Parse Options tab within the Stored Procedures. You can enable this for a specific stored procedure or for all stored procedures.

See Also: For more information on schema object names, see Schema Object Similarities in the *Oracle Migration Workbench Reference Guide for Microsoft SQL Server and Sybase Adaptive Server Migrations*.

How does the Migration Workbench migrate datatypes that the Oracle Server does not support?

Datatypes that the Oracle Server does not support are mapped to Oracle data types that encapsulate similar type characteristics. To view or edit the data type mappings, do one of the following:

- Choose **Tools Menu > Options**, then click the **Data Type Mappings** tab.

See Also: For more information, see the Migration Workbench *Online Help*.

- Access the step within the Capture Wizard that permits you to view and edit the data type mappings.

See Also: For more information, see Step 3: Data Type Mappings in the *Oracle Migration Workbench Online Help*.

- Access the General tab of the property sheet after you have created the Oracle Model to modify column types from the individual table you created or from all tables.

See Also: For more information on data types, see the *Oracle Migration Workbench Reference Guide for Microsoft SQL Server and Sybase Adaptive Server Migrations*.

What do the following error messages mean?

- ORA-00903 - Invalid table name
- ORA-00904 - Invalid column name
- ORA-00942 - Table or view does not exist

These messages can appear in the Migration Workbench UI and error log file while you load the source database into the Migration Workbench or while you create the Oracle Model.

Incomplete Oracle Migration Workbench Repository If you receive one of the previous error messages while you load the source database, you may have an incomplete repository. Incomplete repositories occur when the Migration Workbench is unable to create all of the required tables. This is caused when the tablespace where the migration schema resides is created with incorrect or missing storage options. It can also occur if there are any other problems, such as space limitations, with the tablespace.

To verify a complete repository within the Migration Workbench, make sure that there are 166 tables listed for all available plug-ins. Enter the following:

```
sqlplus migration users/password@database service name select table_name from
user_tables;
```

To resolve an incomplete repository within the Migration Workbench:

1. Create a new tablespace using the default options provided by Oracle Enterprise Manager. Make sure that you have sufficient disk space.
2. In the tablespaces, create a new user and schema.
3. Log into the Migration Workbench as the new user. The Migration Workbench automatically creates the new repository.

You should monitor the `%ORACLE_HOME%\admin\oracle_SID\bdump\alertoracle_SID.log` Oracle Alert log file for the database on an ongoing basis. This log file contains all error messages generated by the destination Oracle database and indicates any tablespace problems.

Incomplete Source Database If you receive one of the preceding error messages while you create the Oracle Model, you may have an incomplete source database. An incomplete source database can occur if you imported the source Microsoft SQL Server database from another Microsoft SQL Server database without importing the master database. The master databases for incomplete source databases are out of sync.

The master database stores all the security information for a Microsoft SQL Server database. This security information, including the ownership of objects, is mapped to the individual databases within the SQL Server. In order to maintain the security information, you must also move the master database. You can resolve an incomplete source database by importing the correct master database into the correct SQL Server.

How are NCHAR, NVARCHAR, NTEXT, or MEMO data types mapped within the Oracle Server?

The Migration Workbench does not support the mapping of NCHAR, NVARCHAR, NTEXT, or MEMO data types from Microsoft SQL Server, Sybase Adaptive Server, or Microsoft Access to the NCHAR, NVARCHAR2, or NCLOB data types in an Oracle database. Instead, the default mapping for these data types is to CHAR, VARCHAR2, and CLOB in Oracle because these Oracle data types support Unicode and multi-byte character sets.

Does the Migration Workbench use SQL*Loader to control files for loading data?

No. The Migration Workbench does not use SQL*Loader control files when loading the data from the source database to the destination Oracle database. However, you can choose the **Object>Generate SQL*Loader Scripts** option to create the SQL*Loader control files in the `%ORACLE_HOME%\omwb\sqlloader_scripts\timestamp` directory.

Can I modify the SQL*Loader control files for loading the data?

Yes. To save a SQL*Loader control file:

1. Use the offline data loading method.
2. Choose the **Object > Generate SQL*Loader Scripts** option within the Migration Workbench.
3. Save the file

You can then update the file at any stage.

Does the data migration process slow down dramatically when migrating tables that contain large volumes of binary data, such as images, binary data, or OLE objects?

Yes. This is a known problem in the data migration component of the Migration Workbench. You can create a data pump using JDBC/ODBC. The %ORACLE_HOME%/jdbc/demo directory contains a demonstration data pump.

You can not use offline data loading for Sybase Adaptive Server and Microsoft SQL Server migrations because SQL*Loader cannot load binary data generated by BCP.

Choose **Object > Generate SQL*Loader Scripts** from the Migration Workbench to use the offline data loading option.

What should I do if I receive the error "ORA-22866: default character set is of varying width" when migrating to a destination Oracle database created with the UTF8 character set?

To overcome this problem, do one of the following:

- Migrate to an Oracle9i database.
- If you are migrating to an Oracle8i database, set the COMPATIBLE initialization parameter as follows:
`compatible=8.1.0.`

What is the meaning of the error message "ORA-01426: numeric overflow" in the Log window?

This error message means that a double value in the source database is outside the allowable range for the Oracle Server. In this case, the Migration Workbench migrates all other columns in the row as normal and migrates a null value for the column that contains the double value. The error message in the Log window indicates the table and the row number affected.

What does the error "Hostdef does not exist" mean?

If you receive this message, you have been disconnected from the server-side processes. This can occur for several reasons. For example, you may have a problem that is associated with a data move. Locate the last table that was moved before the error occurred. Attempt to migrate the table again. If you are disconnected again, report the problem by sending an email message to infomwb_ww@oracle.com.

Why do I receive the "ORA-02298: parent key not found" error message?

You receive this error message if you do not obey the foreign key constraint. When you use the Migration Workbench, you should load the data for the parent table before the data for a referenced table in a foreign key relationship. By loading the reference table data first you create orphaned child records. An Oracle database requires that the column specified in the reference table be either a primary key or unique index. You should make sure that the table has no orphaned child records before issuing an `ALTER TABLE ENABLE CONSTRAINT` command.

See Also: For more information, see Table Design Considerations in the *Oracle Migration Workbench Reference Guide for Microsoft SQL Server and Sybase Adaptive Server Migrations*.

How do client applications access the data that has been migrated from my source database to an Oracle database?

You can use JDBC, Oracle ODBC drivers, Oracle OLEDB, and third-party ODBC and OLEDB drivers to access client applications after you have migrated the source database to an Oracle database. Download the latest version of the Oracle drivers from the OTN Web site at:

<http://otn.oracle.com/software/index.htm>

See Also: For more information on how to extend the client application, see the *Oracle Objects for OLE and OLEDB/ADO Cookbook* and *Using the Oracle ODBC Drivers with Third-Party Products*.

These documents are available on the OTN Web site at:

<http://otn.oracle.com/tech/migration/workbench>

Does the Oracle Server have implicit date conversion from a character string?

No. To use explicit date conversion (TO_DATE), do one of the following:

- Enter the following command within an Oracle session:

```
$ alter session set nls_date_format='dd.mm.yyyy hh24:mi:ss'
```
- Set the NLS_DATE_FORMAT environment variable:

```
$ set NLS_DATE_FORMAT=dd.mm.yyyy hh24:mi:ss
```

Can I migrate a single schema object?

No. The Migration Workbench does not support the migration of individual schema objects.

See Also: For more information on schema objects, see Schema Object Similarities in the *Oracle Migration Workbench Reference Guide for Microsoft SQL Server and Sybase Adaptive Server Migrations* and the *Oracle Migration Workbench Reference Guide for Informix Dynamic Server 7.3 Migrations*.

Can I rename a schema object?

Yes. You can rename tablespaces and users in the Oracle Model. Choose **Object>Rename** or click the right mouse button, then select Rename. Enter the new name of the tablespace or user.

Note: Spaces are represented by underscores (_).

Can I maintain schema object name case-sensitivity when migrating?

An Oracle database stores all schema values in upper-case. Oracle Corporation recommends that you do not maintain case-sensitivity. However, the Migration Workbench enables you to preserve case-sensitivity schema object names. Choose **Tools>Options**, then click the **Create ANSI-compliant names** option from the General tab. Selecting this option creates all schema values using quotes. This ensures case-sensitivity of the tables and columns.

How do I manage case-sensitivity queries?

The Oracle Server evaluates equality in a case-sensitive manner. This means that you can use uppercase values and uppercase fields for comparisons. You can use UPPER functions to simulate case-insensitivity. However, you must apply this to every SQL statement. Oracle9i and Oracle8i have functional indexes so you can manage indexed fields.

Can the Migration Workbench migrate multiple databases?

Yes. You can select the databases that you want to migrate through one of the steps in the Capture Wizard. Choose **Action>Capture Source Database** to access the Capture Wizard.

Does the Migration Workbench allow multi-character field delimiters?

Yes. Oracle9i and Oracle8i allow multi-character field delimiters.

How do I modify the storage options for a table?

To modify the storage options for tables and indexes before running the Migration Wizard:

1. In the Oracle Model, select the Tables container.
2. Go to the Default Storage Options tab of the Tables property sheet.
3. Make the changes that you require and click **Apply**.

Doing this sets the options for all tables in the Oracle Model.

To modify the default creation options for a particular table:

1. Expand the Tables container in the Navigator Pane
2. Select the table that you want to modify.
3. Make the changes that you require in the Override Default Creation Options tab of the property sheet for that table.

Do I need to edit table defaults?

Yes. After migrating to the Oracle Server, you must manually edit the table defaults containing dates and bit-wise operators.

How does the Oracle Server handle temporary tables?

By default, the Migration Workbench uses the temporary tables feature of Oracle9i. When the parser encounters SQL statements such as CREATE #TABLE in a stored procedure, the parser explicitly creates a temporary table. When the parser encoun-

ters SQL statements such as `INSERT INTO #TABLE` in a trigger or stored procedure, it implicitly creates a temporary table. An implicit creation is required when a `CREATE TABLE` statement cannot be found.

These creation (DDL) statements are associated with stored procedures and triggers and are executed by the Migration Wizard before the stored procedures or triggers are created. Clashes can arise when the parser creates multiple copies of the same DDL statement. The clashes are written to the Log window and you can ignore them.

Support

This section contains support questions.

How do I get help on Migration Workbench issues?

You can get help on Migration Workbench from the following sources:

- The Migration Workbench documentation. This consists of the *Oracle Migration Online Help*, an on-line platform-specific reference guide, the Oracle Migration Workbench FAQ, and release notes.
- Technical bulletins. These bulletins are available at the support page of the Migration Workbench. Choose **Help > Online Support** from within the Migration Workbench to obtain an updated list of support facilities.
- The latest discussion forum for the Migration Workbench on the OTN Web site at:
<http://otn.oracle.com/support/bboard/discussions.htm?Forum&10>
- Support information on the Migration Workbench from the OTN Website:
<http://otn.oracle.com/tech/migration/workbench>
- Support by email at infomwb_ww@oracle.com.

How do I know what error messages occurred when running the Migration Wizard?

All error messages are logged to the Log window and the columns of the Log window can be sorted by clicking on the column header. During the creation phase of the migration, the Migration Wizard creates three log files that record all successful CREATE statements and ALTER statements, the equivalent DROP statements and ALTER statements, and all SQL statements that the Migration Workbench was unable to execute. The location of these log files is governed by the log file directory setting in the Logging page of the Options dialog box. The default location is %ORACLE_HOME%\omwb\log. If the Migration Workbench fails to respond, check the contents of the %ORACLE_HOME%\omwb\log\Error . log file to see if there are any error messages recorded. The contents of this file can be sent to the Migration Workbench development team at infomwb_ww@oracle.com. You can also generate reports from the Migration Workbench. This provides you with a formatted list of the error messages.

Informix Dynamic Server Questions

This chapter contains frequently asked questions about using the Oracle Migration Workbench to migrate an Informix Dynamic Server 7.3 database to an Oracle Server. It contains the following sections:

- [Installation and Configuration](#)
- [Data Migration](#)

Installation and Configuration

This section contains Informix Dynamic Server installation and configuration questions.

Are there any schema migration limitations for Informix Dynamic Server?

Yes. The following list includes the schema migration limitations for Informix Dynamic Server:

- Dbspaces
- Database Privileges
- Order Specifying Schema Objects for Migration
- User Privileges
- Defaults
- Roles
- Public Synonyms
- Indexes
- Foreign Key Constraints
- Check Constraints
- Check Constraint Owners

What do I use to connect to an Informix Dynamic Server?

Use Informix Dynamic Server JDBC Driver 1.4 to connect to an Informix Dynamic Server. The Informix JDBC Driver 1.4 is available from the Informix Dynamic Server 7.3 CD.

To connect to an Informix Dynamic Server, you must copy the `ifxjdbc.jar` Informix Dynamic Server JDBC driver file to the `%ORACLE_HOME%\omwb\drivers` directory before running the Migration Workbench. Otherwise, when you attempt to capture a Migration Workbench database, you receive the following error message:

No Suitable Driver Found.

If you receive this error message, copy the `ifxjdbc.jar` file to the `%ORACLE_HOME%\omwb\drivers` directory, then restart the Migration Workbench.

Data Migration

This section contains Informix Dynamic Server data migration questions.

Why do I have problems migrating the Defaults schema when I use the default repository?

If you use the default repository, the Defaults schema objects are not migrated from an Informix Dynamic Server source database to an Oracle destination database. To ensure that schema objects are migrated, select an Oracle9i or Oracle8i repository in the Oracle Migration Workbench Repository Login dialog box.

See Also: For more information on schema object limitations, see Schema Object Similarities in the *Oracle Migration Workbench Release Notes*. The release notes are available from the *Oracle Migration Workbench Online Help* and from the Documentation section of the OTN Web site at:

<http://otn.oracle.com/tech/migration/workbench>

What must I do when the stored procedure parameters that are declared using the LIKE syntax are incorrectly parsed to type CLOB?

This is a known problem with the stored procedure parsing. To avoid it, manually edit the code. You can do this by editing the stored procedures schema object of the Oracle Model. Alternatively, you can edit the stored procedure in the Oracle database after the migration process is complete.

The following table displays the incorrect parsing output and the correct output for the following stored procedure variable definition:

```
create procedure myProc(aParameter LIKE atable.column)
```

Incorrect Parsing	Correct Parsing
<pre>CREATE OR REPLACE PROCEDURE myProc(/* SPCONV-ERR[---]:(REFERENCES TEXT) Converted to CLOB datatype. Insert and Update restrictions apply. */ aParameter CLOB) AS</pre>	<pre>CREATE OR REPLACE PROCEDURE myProc(aParameter aTable.column%TYPE)</pre>

Why does the stored procedure parser fail in the Migration Workbench?

The stored procedure parser can fail if you use hard coded decimal numbers with missing leading or trailing zeros. This is a known problem with the Stored Procedure parser in the Migration Workbench. To prevent it from happening, insert leading or trailing zeros to the numbers in the stored procedure text. You can do this in either in the Informix Dynamic Server database or in the Source Model of the Migration Workbench. The following table shows examples of decimal numbers:

Incorrect Format	Correct Format
0.	0.0
10.	10.0
1024.	1024.0
.32	0.32

Does the Informix Dynamic Server Stored Procedures Parser have any limitations?

Yes. The limitations of the Informix Dynamic Server parser are:

- The parser moves all user-defined comments within the original Informix Dynamic Server stored procedure from their original position to the top of the Oracle stored procedure where they display in a single unit.
- You must place a date format mask in front of any variable or literal that is assigned to a DATETIME column.
- A single Informix Dynamic Server trigger can contain both BEFORE and AFTER constructs. Triggers of this type generate two distinct triggers, a BEFORE and an AFTER trigger, within the Oracle destination database. The stored procedure parser for this release displays code for both of the resulting Oracle destination database triggers in one screen. Access these triggers through the Triggers Property Sheet within the Oracle Model. However, the parser only creates the first trigger in the destination Oracle database. You must create the second trigger manually. To do this, copy the text from the Triggers Property Sheet within the Oracle Model.
- The parser ignores all SET statements, with the exception of SET DEBUG FILE. You might need to manually convert the SET statements.
- The Migration Workbench does not parse the Informix Dynamic Server generic CASE statements, for example CASE WHEN e.sal > 2000 THEN e.sal

ELSE 2000 END CASE. You might need to manually convert these types of statements.

- The in-built Informix Dynamic Server system procedure DBINFO is only partially translated during conversion. You might need to manually convert this procedure.
- The parser ignores all DDL statements within stored procedures, with the exception of the CREATE TABLE statement.
- Informix Dynamic Server enables you to use keywords as identifiers within stored procedures. However, you cannot use all Informix Dynamic Server keywords as identifiers.

See Also: For more information on the keywords that you cannot use, see the *Oracle Migration Workbench Reference Guide for Informix Dynamic Server 7.3 Migrations*.

Does the Migration Workbench fully support the migration of LOB data using SQL*Loader scripts?

No. The Migration Workbench does not fully support the migration of LOB data using SQL*Loader scripts.

The unload command dumps binary data as a series of hexadecimal characters. Within a SQL*Loader control file, you can apply the HEXTORAW SQL function to this field of hexadecimal character data to produce binary data. The Migration Workbench can then write binary data to a LONG RAW column. The Migration Workbench will generate a control file to do this automatically.

You can use the Migration Workbench to generate the SQL*Loader control file for the table which stores binary data.

To generate the SQL*Loader control file:

1. After mapping the source database to the Oracle database with the Oracle Migration Workbench, navigate to the table in the Oracle Model, right-click on the icon for the table, and select **Generate SQL*Loader Scripts**. This generates the Informix Dynamic Server extraction script and the Oracle SQL*Loader script and control file for the table.
2. In the SQL*Loader control file generated by the Oracle Migration Workbench, verify that the maximum CHAR size is large enough for the dumped hexadecimal data, in the line for the column which stores the binary data.

Because the column in the Oracle database table is defined as **LONG**, you must use a length specifier on the **CHAR** keyword to specify a maximum length. The maximum length is 2 gigabytes for **LONG** datatypes. Doing this guarantees that a large enough buffer is allocated for the value.

Note: Ensure that the **CHAR** maximum length specifier is set high enough to hold the hexadecimal representation for the largest binary data field stored in the source table (for example, twice the number of bytes for the image).

The logical records being loaded must completely fit into the available memory on the system running **SQL*Loader**.

Microsoft Access Questions

This chapter contains frequently asked questions about using the Oracle Migration Workbench to migrate from a Microsoft Access 2.0, 95, 97, or 2000 database to an Oracle Server. It contains the following sections:

- [Pre-Installation](#)
- [Installation and Configuration](#)
- [Data Migration](#)

Pre-Installation

This section contains Microsoft Access pre-installation questions.

Does the Migration Workbench support Microsoft Access 2000 migration to Oracle?

Yes. The Migration Workbench release 9.2.0 supports the migration of Microsoft Access 2000 to a destination Oracle database. In order to migrate a Microsoft Access 2000 database, you must have Microsoft Access 2000 installed on your system.

See Also: See the following question for more information .

How do I migrate a Microsoft Access database to an Oracle database?

Before you can migrate a Microsoft Access database to an Oracle database, you must first export the MDB file to an XML file. Use the Oracle Migration Workbench Exporter for Microsoft Access to do this. This is located in the %ORACLE_HOME%\omwb\msaccess_exporter directory.

The Oracle Migration Workbench Exporter extracts the schematic information from the MDB file and stores it in the XML file. The Migration Workbench then uses this XML file to capture the Microsoft Access database within the Capture Wizard.

If the source Microsoft Access database uses linked tables, the schematic information for the attached MDB files are also automatically stored in the XML file.

You can export and capture only one Microsoft Access database at a time.

Installation and Configuration

This section contains Microsoft Access installation and configuration questions.

What software do I need to migrate a Microsoft Access 97 database to an Oracle database?

In order to migrate a Microsoft Access 97 database to an Oracle database, you must have the following software installed on the same system as the Migration Workbench:

- An ODBC driver release 4.00.4202.00 or greater.

You can obtain this by downloading the latest version of the Microsoft Data Access Components (MDAC) from the Microsoft Web site at:

<http://www.microsoft.com>

- Microsoft Data Access Objects (DAO) release 3.5.

You can obtain this by downloading Microsoft Access 97.

- Microsoft Access 97.
- An Oracle ODBC driver.

This allow Microsoft Access forms and reports to work with the migrated data. To install the Oracle ODBC driver, select the **Custom Install** option when installing the Migration Workbench. After you have installed the Oracle ODBC driver, you should set the Oracle ODBC Data Source.

See Also: For more information, see Setting the Oracle ODBC Data Source in the *Oracle Migration Workbench Online Help*.

What software do I need to migrate a Microsoft Access 2000 database to an Oracle database?

To migrate Microsoft Access 2000 database to an Oracle database, you must have the following software installed on the same system as the Migration Workbench:

- An ODBC driver release 4.00.4202.00 or greater. You can obtain this by downloading the latest version of the Microsoft Data Access Components (MDAC) from the Microsoft Web site at:
<http://www.microsoft.com>
- Microsoft Data Access Objects (DAO) release 3.6. You can obtain this by downloading Microsoft Access 2000.
- Microsoft Access 2000.
- You must install the Oracle ODBC driver to enable Microsoft Access forms and reports to work with the migrated data. To install the Oracle ODBC driver, select the **Custom Install** option when installing the Migration Workbench. After you have installed the Oracle ODBC driver, you should set the Oracle ODBC Data Source.

See Also: For more information, see Setting the Oracle ODBC Data Source in the *Oracle Migration Workbench Online Help*.

Note: You must install the Oracle ODBC driver version 9.0.1

What happens if the following message appears when you log into the Oracle

Migration Workbench: "There is no Microsoft Access ODBC driver software installed on this system. Please install the software before using the Microsoft Access plugin."

If the Microsoft Access ODBC driver is installed correctly, this issue may occur because you are logged on to a system where another user (with different user privileges) has installed the Microsoft Access Database.

To resolve this issue you must have the same privileges as the other user.

Data Migration

This section contains Microsoft Access data migration questions.

Why does the Oracle Model hang when I map foreign keys?

If you have defined a foreign key relationship between two tables outside of the MDB file that contains these tables, such as between two linked tables, the Oracle Model might hang. To avoid this problem, define the table relationship between tables within the MDB file.

What should I do if I receive the "ORA-00001: Unique constraint violated message displayed in error.log or as an alert" error message?

If you receive the ORA-00001 error message, the Migration Workbench hangs while it loads indexes.

You might receive this error if there is a problem with the owner profile within the Migration Workbench Repository or if the administrator of the Microsoft Access database does not have full write permissions. You may also receive this error message if the PL/SQL package has not been created properly.

Granting Permissions The ORA-00001 error message can appear if the Microsoft Access database administrator does not have full permissions. To fix this problem:

1. Check the security of the administrator in Microsoft Access by choosing **Tools > Security > User and Group Permissions**.
2. Make sure that all of the permission options are selected for the administrator.

Creating PL/SQL Packages If the PL/SQL package was not created correctly in the Oracle database, no triggers associated with the Microsoft Access Source Model tables work. The Source Model uses sequence and trigger pairs to generate unique IDs for some of the columns within the tables.

To check if the PL/SQL package was created properly, log in to SQL*Plus and execute the following **SELECT** statement:

```
select tblid from acc_table;
```

If all values returned by this statement are -1, the trigger that the Migration Workbench uses to return unique table IDs was not created.

You can test if the Migration Workbench correctly created the triggers by executing the following statement:

```
select trigger_name from all_triggers where table_name in ('ACC_DB_INFO', 'ACC_
TABLE');
```

If the following names are not returned by this statement then the triggers were never created. If this is the case, consult with the DBA and investigate why triggers were not created.

```
ACC_TR_DB_INCR
ACC_TR_TBL_INCR
```

If the previous names are returned, you should verify that the triggers work. The **ACC_DB_INFO** table has a sequence/trigger pair defined on the **DBID** column. To verify that the triggers work:

1. Execute the following statement several times:

```
insert into ACC_DB_INFO values (999,'f',12,'d','a','s',1,1,1);
```

2. Perform a select of the **DBID** value in the **ACC_DB_INFO** table to ensure that the values in the **DBID** column are sequential.

If the trigger does not work, the value specified in the **INSERT** statement (in our example 999) for the **DBID** column is displayed.

To test whether you can manually create the sequence/trigger pair execute the following commands:

```
DROP SEQUENCE ACC_SEQ_DB_INCR;
CREATE SEQUENCE ACC_SEQ_DB_INCR START WITH 1;
CREATE OR REPLACE TRIGGER ACC_TR_DB_INCR
BEFORE INSERT ON ACC_DB_INFO
FOR EACH ROW
BEGIN
SELECT ACC_SEQ_DB_INCR.NEXTVAL INTO :NEW.DBID FROM DUAL;
END;
/
```

Consult the DBA if you receive any error messages when attempting to execute the previous commands.

After you have attempted to manually create the sequence and trigger, the following error might display:

ORA-06554 package DBMS_STANDARD must be created before using PL/SQL

This error indicates that there is something wrong with the way in which you set up the database. To correct this, create a new database instance using the database configuration assistant that ships as standard with Oracle9i or Oracle8i.

What if I receive one of the following messages while attempting to launch the Oracle Migration Workbench Exporter for Microsoft Access:

- Prompted to associate a file type with the omwb.mde file
- Incorrect version of Microsoft Access is executed
- Error: ActiveX component can't create object
- Error: Unable to convert or enable MDE file.

The version of Microsoft Access must be compatible with the Microsoft Access database that you want to migrate. If you receive one of the preceding error messages, you might have a version of Microsoft Access that is incompatible with the Microsoft Access database that you are migrating. If you have more than one version of Microsoft Access installed on the same system, the incorrect version of the Oracle Migration Workbench Exporter for Microsoft Access might execute.

To ensure that you use the correct version of the Oracle Migration Workbench Exporter for Microsoft Access (omwb.mde) on the system:

1. Start the version of Microsoft Access associated with the database that you would like to migrate to a destination Oracle database. For example, if you are migrating a Microsoft Access 2000 database, open Microsoft Access 2000. If you are migrating any other Microsoft Access database, open Microsoft Access 97.
2. Choose **File > Open Database**.
3. From the Files of Type option, select **Microsoft Access Databases (*.mde)**.
4. Click **Advanced**.
5. From the Look In option, search for the omwb97.mde or omwb2000.mde files located in the %ORACLE_HOME%\Omwb\msaccess_exporter directory.

Why does the Migration Workbench not handle validation rules with the following syntax:

- between x and y
- is null

For validation rules, the Migration Workbench generates triggers. The Migration Workbench does not parse the between x and y or is null syntax correctly. In order to overcome this problem, the between x and y or is null syntax, replace the trigger text in the Oracle Model. The following table shows the incorrect syntax and the correct syntax that you should use to edit the code of the trigger.

Incorrect Syntax	Correct Syntax
IF NOT (:new.column>new.column=x AND:new.column=y AND:new.column<=:new.column is Null) THEN	IF NOT (:new.column>=x AND:new.column<=y OR:new.column is Null) THEN

Can I migrate my Microsoft Access security settings to an Oracle Server?

No. The Migration Workbench does not support migrating Microsoft Access databases that have security enabled. To ensure that the Migration Workbench can migrate the Microsoft Access table data, copy the contents of the secured database into a new database. Everything is copied to the new database, except the security settings. You can then migrate the new database to an Oracle Server.

To copy the contents of the secured database into a new database:

1. In Microsoft Access, choose **File > New Database**.
2. Select the **Blank Database** icon and click **OK**.
3. From the File New Database dialog box, enter a name for the database and click **Create**.
4. From within the new database, choose **File > Get External Data > Import**.
5. Select the secured Microsoft Access database that you want to import and click **Import**.

6. From the Import Objects dialog, click **Options**. Ensure that the Relationships and Definition and Data options are selected.
7. From the Tables tab, choose **Select All**.
8. Click **OK**.

How do I avoid erroneous relations within Microsoft Access 95?

There is a bug within Microsoft Access 95 that sometimes generates erroneous relations. Since these erroneous relations do not appear in the Microsoft Access IDE user interface, you cannot delete them by using the application because they are hidden from the user. To correct this problem, create a new copy of the Microsoft Access database with no relations specified. You can then manually re-create the correct relations in the new database.

To copy the contents of the Microsoft Access database without specifying relations:

1. Create a new empty Microsoft Access database by choosing **File > New Database**.
2. Select **Blank Database** and click **OK**.
3. From the File New Database dialog box, type a name for the database and click **Create**.
4. From within the new database, choose **File > Get External Data > Import**.
5. Select the original Microsoft Access database that you want to import.
6. From the Import Objects dialog, select **Options**.
7. Deselect **Relationships** from the Import section, then click **OK**.

Note: If you want to migrate the legitimate relations from Microsoft Access, you must manually re-create them within the new database.

8. From the Tables tab, choose **Select All**.
9. Click **OK**.

Note: If you want to migrate the legitimate relations from Microsoft Access, you must manually re-create them within the new database.

Can I migrate a replica Microsoft Access database?

No. The Migration Workbench does not permit migration of replica Microsoft Access databases.

Can I use the Migration Workbench to migrate multiple Microsoft Access databases to a single Oracle user?

Yes. To migrate multiple Microsoft Access databases to a single Oracle user, give all of the Microsoft Access databases the same name. All of the Microsoft Access databases must have the same name because the Migration Workbench uses the name of the Microsoft Access MDB file as the user name for the destination Oracle user.

Ensure that all the Microsoft Access databases with the same name reside in different directories from each other. Then, use the Capture Wizard to select the databases for a simultaneous migration.

Microsoft Access databases that have been captured in the Source Model appear individually, however, they all appear with the same name. You can identify a database by viewing its path in the Source Model property sheet. When you view the Oracle Model, you see that there is only one user. This user contains all schema objects from the multiple Source Model databases.

You can have a Microsoft Access application database that contains linked tables to another Microsoft Access database. To migrate multiple Microsoft Access databases containing attached tables to a single Oracle user:

1. Rename the databases to the name of the Oracle user where you want to migrate, for example `employee.mdb`.
2. Open the first `employee.mdb` file in Microsoft Access, then choose **Tools > Add-Ins > Linked Table Manager** to refresh the links.
3. From the Migration Workbench, choose the **Action > Capture Source Database** option to start the Capture Wizard.
4. Within one of the steps of the Capture Wizard you are asked to add an Microsoft Access database. Select the first `employee.mdb` file, then proceed with migration.
5. Use the Capture Wizard to migrate the next `employee.mdb` file.

Note: If schema objects of the same type have the same name in multiple Microsoft Access databases the Migration Workbench automatically detects and resolves this naming conflict.

See Also: For more information on Microsoft Access architecture, see the *Microsoft Access Reference Guide*.

Can my hyperlinks work after I migrate my database to an Oracle Server?

No. Oracle Servers do not support hyperlinks. Therefore, the Migration Workbench simply brings over the raw contents of the column.

What does the message "No primary key defined on table name; you are unable to update records after migration." mean?

This error message occurs because the specified table does not have a primary key defined on it. Jet requires a primary key on tables in Oracle databases in order to support dynasets against those tables. If the Oracle database table does not have a primary key, Jet only opens a non-editable snapshot on the table. It is possible to define a table in Microsoft Access that you can update yet does not have a primary key. When you migrate the table to an Oracle database, it does not have a primary key; therefore, the Jet engine is unable to update it. If you want to update the contents of the table through the Jet engine, do one of the following:

- Ensure that the original Microsoft Access table has a primary key defined on it. You then must re-capture the Microsoft Access database.
- Define a primary key on the migrated Oracle database table before you use the Migration Workbench to modify the Microsoft Access database.

See Also: For more information on Microsoft Jet error messages, see the *Microsoft Access Reference Guide*.

What should I do if I encounter the error message "JET/DAO Error 3050: Couldn't lock file"?

When DAO opens a Microsoft Access database, it automatically generates a LDB file. This error indicates that the LDB file is read-only and cannot be updated. To overcome this error, you should make the LDB file writable by altering the properties of the file and then reselect the Microsoft Access database for migration.

See Also: For more information on Microsoft Jet error messages, see the *Microsoft Access Reference Guide*.

Why must I define an Oracle ODBC data source when migrating from Microsoft Access?

You must define an Oracle ODBC data source to modify the Microsoft Access database. Defining an Oracle ODBC data source allows you to continue using the Microsoft Access forms and reports after you have used the Migration Workbench to migrate the data to a destination Oracle database.

Why must I modify the Microsoft Access database?

To continue using the Microsoft Access front end with the data that you have migrated to an Oracle Server, the Migration Workbench must make some modifications to the Microsoft Access database.

To modify the Microsoft Access database, you must rename local tables. However, during the data move phase of the migration, the Migration Workbench references the original table names and expects these names to exist in the Microsoft Access database. Therefore, you should modify the Microsoft Access database after you have successfully migrated the data to Oracle.

Does the Migration Workbench support offline data loading in Microsoft Access?

Yes. The Migration Workbench uses SQL*Loader to provide an offline data loading capability, which improves the migration of large tables from Microsoft Access. To use offline data loading, you must manually create the data files from within the Microsoft Access Integrated Development Environment (IDE).

Note: Microsoft Access 97 only supports the creation of a data file on a table-by-table basis. Therefore, you must perform the steps outlined in this section for each Microsoft Access table individually.

The base directory for the SQL*Loader script output for Microsoft Access is %ORACLE_HOME%\0mwb\sqlloader_scripts\MSAccess*timestamp*. The default Log directory is %ORACLE_HOME%\0mwb\sqlloader_scripts\. The Migration Workbench creates a directory in this base directory that represents the date and time that you generated the SQL*Loader scripts. For example, a sub-directory called 1-10-1999_17-58-16 indicates that the scripts were generated at 17:58 p.m. on 1st October, 1999.

To manually create the data files from within the Microsoft Access Integrated Development Environment (IDE):

1. Open the Microsoft Access database in the Microsoft Access environment.
2. Select the table whose data you want to export to a data file.
3. Choose **File > Save As/Export**.
4. Choose the **To an External File or Database** option, then click **OK**.
5. Choose the **Text Files** from the Save as type box. Make sure that the **Save Formatted** option is not selected, then click **Export**.
6. Select the **Delimited - Characters comma or tab separate each field** option, then click **Next**.
7. Click **Other** as the delimiter that separates the fields, then insert the § character.

Note: The § character is used by the SQL*Loader control file that is generated by the Migration Workbench. You can enter § by pressing ALT+0167.

8. Select **{none}** from the Text Qualifier box, then click **Next**.
9. Enter the following path to export the file.
%ORACLE_HOME%\0mwb\sqlloader_scripts\MicrosoftAccess_timestamp directory.

Note: Make sure that the name of the file is identical to the name of the table and that the extension you specify for the data file is DAT. This is the naming convention used by the SQL*Loader control file generated by the Migration Workbench.

10. Click **Finish** to generate the data file, then click **OK**.
11. Now that you have manually created the data files, you can run the sql_load_script.bat file in the %ORACLE_HOME%\0mwb\sqlloader_scripts\MicrosoftAccess_timestamp directory.

How do I migrate the Microsoft Access Memo columns in tables?

The Oracle ODBC driver automatically handles both BLOB and CLOB columns. By default, all Microsoft Access Memo fields are mapped to CLOB datatypes in an Oracle database. You can use the Microsoft Access front end with the destination Oracle database. Unlike LONG columns, from Oracle 8.x onwards you can have any number of LOB columns in a table. You can have either internal LOBs, meaning they

are inside the database, or external LOBs, meaning they are in a file on the file system but under the database's transactional control. The advantages of LOBs are that you can replicate them or index them using the context option.

What should I do if the following error is reported during data load from Microsoft Access to Oracle: "EXCEPTION: LoadTableData.run() : [Microsoft][ODBC Microsoft Access Driver] '(unknown)' is not a valid path. Make sure that the path name is spelled correctly and that you are connected to the server on which the file resides"?

This error is associated with the internal ODBC DSN which is installed with the Migration Workbench. The most likely cause is that the Microsoft Access plug-in has been reinstalled into a different Oracle Home or that the small.mdb file installed with the Migration Workbench has been deleted or moved.

To resolve this issue:

1. Choose **Start > Settings > Control Panel > ODBC Data Sources > System DSN**
2. Double-click **Microsoft Access DSN**.
3. Ensure that the **Database** path name is spelled correctly, and that this path points to a valid .mdb file.
4. If the path is not correct choose **Select** and browse to a valid mdb file location.

Microsoft SQL Server and Sybase Adaptive Server Questions

This chapter contains frequently asked questions about using the Oracle Migration Workbench to migrate from a Microsoft SQL Server 6.5, 7.0, 2000, or Sybase Adaptive Server 11, 12 database to an Oracle Server. It contains the following sections:

- [Pre-Installation](#)
- [Data Migration](#)

Pre-Installation

This section contains Microsoft SQL Server and Sybase Adaptive Server pre-installation questions.

Does the Migration Workbench support Microsoft SQL Server 6.0? If not, are there any plans to support this version of Microsoft SQL Server?

No. The Migration Workbench does not support Microsoft SQL Server 6.0. The Capture Wizard checks the version of the source database to ensure that it is Microsoft SQL Server 6.5 or Microsoft SQL Server 7.0. There are no plans to support Microsoft SQL Server 6.0.

Can I migrate from Microsoft SQL Server or Sybase Adaptive Server to an Oracle7 database using the Migration Workbench?

Yes. However, the Migration Workbench does not officially support migration to Oracle7. To migrate from Microsoft SQL Server or Sybase Adaptive Server to an Oracle7 database, the tables of the source database must not contain multiple text or binary columns.

LOBs are not available in Oracle7 so BLOB and CLOB datatype mappings do not work in migrations to Oracle7. However, you might be able to use LONG and LONG RAW datatypes.

Before capturing the source database, you must modify the datatype mappings. If the size of the text is less than 2000 characters in Oracle7 or 4000 characters in Oracle8, you should map the TEXT datatype to the VARCHAR2 datatype. Otherwise, if the text is greater than 4000 characters, map the TEXT datatype to the LONG datatype. You should also map the following datatypes to the LONG RAW datatype:

- IMAGE
- BINARY
- VARBINARY

It is possible to have only one LONG or LONG RAW column in an Oracle7 table. Therefore, the source table must only contain one of these data types. You must switch temporary tables off in the parser.

What is the meaning of the "Owner name is NULL. Check the MASTER database is correctly recovered" message?

If you receive either of these messages, the master database might not be in sync with the database you are migrating. SQL Server and Sybase have their own master

database which stores all of the security information. This security information, including the ownership of objects, is mapped to the individual databases within SQL Server and Sybase. In order to ensure that this security information is not lost, you must migrate the master database.

Why must I define a SQL Server or Sybase System 11 ODBC data source?

You must define a SQL Server or Sybase System 11 ODBC data source so that the Migration Workbench can capture the data dictionary and extract the data from the Microsoft SQL Server or Sybase Adaptive Server database you want to migrate.

Why does a SELECT statement on a temporary table fail when I have migrated a stored procedure to a destination Oracle database?

A SELECT statement on a temporary table (#TABLES) might fail when a SESSID column is added. For example, the following SQL statement does not work in a destination Oracle database:

```
select SESSID, * from a_table
```

However, the following SQL statement does work in an Oracle database:

```
select SESSID, tablea.* from a_table
```

Can you migrate roles and privileges using the Migration Workbench?

Yes. Every Microsoft SQL Server or Sybase Adaptive Server user is mapped to its associated login and it is the login that is created in the destination Oracle database. The Migration Workbench automatically creates Oracle roles for each Microsoft SQL Server or Sybase Adaptive Server group and then assigns that role to each Microsoft SQL Server or Sybase Adaptive Server login. That is, a user is created in the Oracle Server for the login, a role is created for the group, and the user is assigned the role.

Every privilege that a Microsoft SQL Server or Sybase Adaptive Server user has is also given to its equivalent Oracle user. Consider the following example:

- LoginA is associated with userA in the PUBS database.
- LoginB is associated with userB in the PUBS database.
- The userA owns the EMP table and grants userB select permission on that table.

In this example, the Migration Workbench does the following:

- Creates two users: loginA and loginB in the Oracle Server.

- Assigns loginA ownership to the table EMP.
- Grants loginB select access on the table.
- Creates private synonyms for every object in another schema where there is user privilege.

In this example, loginB has a private synonym called EMP. This synonym is really loginA.emp.

See Also: For more information, see the Roles and Privileges sections in the *Oracle Migration Workbench Reference Guide for Microsoft SQL Server and Sybase Adaptive Server Migrations*.

How does the Migration Workbench map the Microsoft SQL Server database?

Microsoft SQL Server is one server with many databases, while an Oracle Server is one instance with many tablespaces. Database names are mapped to tablespaces. If you want the schema to have the same name as the Microsoft SQL Server database, then you can rename the user in the Oracle Model. Optionally, you can rename the user to the same name as the database name.

How are Microsoft SQL Server and Sybase Adaptive Server user-defined types handled?

The Migration Workbench maps Microsoft SQL Server and Sybase Adaptive Server user-defined types to their base types. The Migration Workbench finds out what the base type is, then maps that to the equivalent destination Oracle database using the data type mappings.

See Also: For more information, see the User-Defined Data Types section in the *Oracle Migration Workbench Reference Guide for Microsoft SQL Server and Sybase Adaptive Server Migrations*.

What happens if I receive the following error message: "Failed to load Source Model [INTERSOLV] [IDBC SQL Server] [SQL Server] WARNING! Some characters could not be converted into client's character set. Unconverted bytes were changed to question marks (?)" ?

This error may occur because the correct ODBC translator has not been chosen when the Sybase Adaptive Server ODBC driver was configured.

To resolve this issue select the correct ODBC translator:

1. Choose **Start > Settings > Control Panel > ODBC Data Sources > System DSN**

2. Double -click **Sybase DSN**
3. Choose **Advanced**
4. Select the required **ODBC translator** from the list and begin the capture again.

Data Migration

This section contains Microsoft SQL Server and Sybase Adaptive Server data migration questions.

Does the Migration Workbench support offline data loading in Microsoft SQL Server or Sybase Adaptive Server data migration?

Yes. The Migration Workbench uses Microsoft SQL Server BCP or Sybase Adaptive Server BCP in conjunction with SQL*Loader to provide offline data loading for large tables.

To create the BCP data extraction script and the SQL*Loader control files for all tables:

1. Click the **Tables** folder in the Oracle Model.
2. Choose the **Object > Generate SQL*Loader Scripts** option.

Note: You can also generate the scripts for a specific table by highlighting that table in the Oracle Model, then following the steps outlined for all tables.

For illustration purposes, this section refers to the Microsoft SQL Server 6.5 plug-in. All data extraction scripts are stored in the following directory:

script_directory = %ORACLE_HOME%\0mwb\sqlloader_scripts

The base directory for the SQL*Loader script output for Microsoft SQL Server 6.5 is *script_directory*\SQLServer6\.

The Migration Workbench creates a directory in the base directory that represents the date and time that you generated the SQL*Loader scripts. For example, a sub-directory called 1-10-1999_17-58-16 indicates that the scripts were generated at 17:58 p.m. on 1st October, 1999.

A BCP extraction script called *bcp_extract.bat* is created in the *script_directory*\SQLServer6\1-10-1999_17-58-16 directory.

As part of the Generate SQL*Loader Script command, a sub-directory called `Oracle` is created in the `script_directory\SQLServer6\1-10-1999_17-58-16` directory. The `Oracle` directory contains SQL*Loader control files and a SQL*Loader script called `sql_load_script.bat`. The SQL*Loader control files expect the data files that are created to be located in this directory also. Therefore, you should copy the `bcp_extract.bat` file into the `Oracle` directory before you execute it.

To use these scripts to execute the manual data extraction:

- 1. Add the server name and password for the server where the Microsoft SQL Server database is located to the `bcp_extract.bat` file:

```
bcp pubs_ro.dbo.employee out EMPLOYEE.dat -c -t "<ec>" -r "<er>" -Usa -Psa
-Sientsrv9
```

The following table describes the syntax of the preceding example:

Code	Description
pubs_ro.dbo.employee	The table you are extracting the data from is the employee table, owned by the <code>dbo</code> user located in the <code>pubs_ro</code> database on <code>ientsrv9</code> .
out EMPLOYEE.dat	The file where the data extracted from the employee table is saved.
-c	Extracts all data in character mode.
-t "ec"	Uses <code><ec></code> as the column terminator.
-r "er"	Uses <code><er></code> as the row terminator.
-Usa	User name. You should log on to Microsoft SQL Server using the <code>sa</code> user name.
-Psa	Password. You should set the password for the <code>sa</code> user name to <code>sa</code> .
-Sientsrv9	Location of the Microsoft SQL Server system. In this case it is <code>-Sientsrv9</code> .

- 2. Run the `bcp_extract.bat` file to generate the data files. This extracts the data from the tables into data files called `table_name.dat`.
- 3. Move the data files into the `script_directory\SQLServer6\1-10-1999_17-58-16\Oracle` directory.

4. If the destination Oracle database does not reside on the same system as the Migration Workbench, you should FTP the entire *script_directory\SQLServer6\1-10-1999_17-58-16\Oracle* directory to the system where the destination Oracle database is located.
5. Execute the *sql_load_script.bat* file to insert the data from the data files into the equivalent Oracle tables.

Note: The scripted data move does not support BLOB migration. You can remove the `\n` character if it exists in CLOBs.

How do I run BCP from a client system?

You must generate all of the BCP and SQL*Loader scripts that are required to move the data by doing one of the following:

- Choose **Object > Generate SQL*Loader Scripts** from within the Migration Workbench.
- Enter the following lines where *database name* is the server name where the Microsoft SQL Server database is located:

```
bcpl pub_ro.dbo.pub_info out PUB_INFO.dat -c -t "ec" -r "er" -Usa -Psa  
-Sdatabase name
```

For example, if the Microsoft SQL Server database is running on server *ientsrv9*, replace *-Sdatabase name* with *-Sientsrv9*. This allows BCP to run the client server.

To run BCP from a client system:

1. Create a migration user in the destination Oracle database.
2. Log into the Migration Workbench as the migration user.
3. Choose **Action>Capture Source Database** to run the Capture Wizard and map the database.
4. From the Oracle Model, choose **Object>Generate SQL*Loader Script**.

Are there any guidelines for how long it takes to convert high volumes of data from Microsoft SQL Server or Sybase Adaptive Server to an Oracle Server using the Migration Workbench?

No. Performance tests have not been carried out on the data migration process of the Migration Workbench. However, it is known that the setup affects performance.

For example, if the Oracle Server and Microsoft SQL Server or Sybase Adaptive Server are not located on the same server, network bandwidth has an affect on performance.

Presently, data migration involves reading the data from the source Microsoft SQL Server or Sybase Adaptive Server database through a JDBC / ODBC bridge and writing that data to an Oracle database through JDBC. It is a straightforward data pump. However, the Migration Workbench takes advantage of array inserts and 'commit points' to improve performance. You can use the BCP scripts in conjunction with SQL*Loader to provide offline data loading for non-binary data.

Data Types

The following questions are about SQL Server and Sybase Adaptive Server datatypes.

How are TIMESTAMP data-types migrated?

The Migration Workbench automatically migrates TIMESTAMP columns by creating a NUMBER column where an UPDATE/INSERT trigger is defined. A SEQUENCE is also defined from where the trigger retrieves the next integer value, for example last value +1, whenever it fires.

What is the Microsoft SQL Server or Sybase Adaptive Server DATETIME data type mapped to within an Oracle database?

The Microsoft SQL Server or Sybase Adaptive Server DATETIME data type is mapped to the DATE data type in Oracle. Unlike the DATETIME data type in Microsoft SQL Server or Sybase Adaptive Server, the DATE data type in Oracle does not support milliseconds. This can result in a loss of precision.

How does the Migration Workbench handle Microsoft SQL Server 7.0 VARCHAR data types greater than 4000?

The Migration Workbench converts any column that contains a VARCHAR field with a length greater than 4000, to a CLOB column in an Oracle database. The CLOB column provides a very efficient method of storing big text columns.

Parsing

The following questions are about SQL Server and Sybase Adaptive Server parsing.

What is the equivalent of the Microsoft SQL Server or Sybase Adaptive Server

SYSOBJECTS table in an Oracle database?

The ALL_OBJECTS table in an Oracle database contains information on all objects that reside in the database and that exist in the SYSOBJECTS table.

Can I edit a stored procedure, trigger, view, or check constraint in the Source Model?

Yes. You can edit the text for any of these schema objects and then parse them. To parse a stored procedure, highlight it and choose **Object>Parse**.

See Also: For more information, see the Triggers and Stored Procedures section in the *Oracle Migration Workbench Reference Guide for Microsoft SQL Server and Sybase Adaptive Server Migrations*.

How are Microsoft SQL Server and Sybase Adaptive Server EXEC statements parsed?

The Migration Workbench assumes that SQL statements, such as EXEC (exec_string) are dynamic SQL and are sent to an EXEC_SQL PL/SQL stored procedure for execution.

Because the contents of the exec_string string are not parsed, you might need to manually edit this string. Other EXEC statements such as EXEC (exec_string) are converted to PL/SQL procedure calls.

Can I use double brackets (()) in T/SQL statements?

No. You cannot use double brackets (()) in T/SQL statements because the parser does not convert them. The extra brackets are not required. The following is an example of valid and invalid T/SQL statements:

Valid Statement	Invalid Statement
SELECT @TIENE = (SELECT COUNT (*) FROM PRE_VEICULOS)	SELECT @TIENE = ((SELECT COUNT (*) FROM PRE_VEICULOS))

Does the Migration Workbench parser support cast operations?

No. The the Migration Workbench parser does not support cast operations. However, instead of using a CAST statement, you can use a CONVERT statement, for example:

Valid CONVERT Statement	Invalid CAST Statement
convert (varchar(20), @un_calor_iva)	cast (@un_valor_iva as varchar(2))

Can I use the LIKE clause in SQL statements?

Yes. You can use LIKE in most instances where UNIX-style regular expressions such as square brackets [] are not used. However, you cannot use the following construct outside a SQL statement:

IF (expression) LIKE (expression)

Can I migrate subqueries in the SELECT clause of Microsoft SQL Server and Sybase Adaptive Server?

No. An Oracle Server does not allow you to completely contain subqueries in the SELECT clause of another query. An Oracle Server has stricter requirements for explicit naming and the location of the subquery than Microsoft SQL Server and Sybase Adaptive Server. For example, Microsoft SQL Server allows the following syntax, but an Oracle Server does not:

```
select TABLE1.FIELD1, (select count(*) from TABLE2)
from TABLE1
where <some criteria
```

Instead, an Oracle Server allows the following syntax:

```
select TABLE1.FIELD1, TABLEALIAS.COUNTALIAS
from TABLE1, (select count(*) as COUNTALIAS from TABLE2) TABLEALIAS
where <some criteria
```

Does an Oracle Server support full UNIX-style regular expression handling?

No. An Oracle Server does not support full UNIX-style regular expression handling.

Are DUMP TRANSACTION and DROP INDEX SQL commands supported?

No. LOAD and DUMP commands are not supported in stored procedures. The Migration Workbench ignores most Microsoft SQL Server and Sybase Adaptive Server DDL commands and writes a warning to the Log window within the Migration Workbench.

How do I avoid problems with cyclic procedures?

If you place the cyclic procedures that you reference in a package, you can compile the cycle procedure because packages consists of a declaration section and a body section. The declaration and body is similar to the C programming language header .h files and implementation .c files. The following lines are an example of packaging:

```
function a(myin INTEGER) return integer;
function b(myin INTEGER) return integer;
function c(myin INTEGER) return integer;
end;
create or replace package body allthree as
function a(myin INTEGER)
RETURN INTEGER
as
    myint INTEGER:=myin;
    BEGIN
    if (myin<3)
    then
    myint:=allthree.b(myint+1);
    end if;
    return myint;
END a;
function b(myin INTEGER)
RETURN INTEGER
as
    myint INTEGER:=myin;
    BEGIN
    if (myin<3)
    then
    myint:=allthree.c(myint+1);
    end if;
    return myint;
END b;
function c(myin INTEGER)
RETURN INTEGER
as
    myint INTEGER:=myin;
    BEGIN
    if (myin<3)
    then
    myint:=allthree.a(myint+1);
    end if;
    return myint;
```

```
END c;
```

Does the Migration Workbench support the conversion of T/SQL stored procedures that return multiple result sets?

The Migration Workbench creates a single cursor variable in an OUT argument in the Oracle Server and uses that variable for all the converted result set SELECT statements. This causes only the results of the last SELECT statement that was opened to be returned in the cursor variable. The results of all other SELECT statements are lost.

To prevent the loss of SELECT statement results, create a separate cursor variable argument for each result set SELECT statement and open each cursor variable using a separate SELECT statement. Alternatively, you can use Oracle packages.

Before you can use Oracle packages to prevent the loss of SELECT statement results, you must comment out all result sets to ensure that the Migration Workbench does not attempt to parse them:

```
create procedure test
as
declare
  @x varchar(10)
select @x = "[Customer]"
select @x /*result set*/
select * from ln_display where ptid >1 /*result set with multiple rows*/
```

To implement the work-around using Oracle packages:

1. Create a complete Oracle package in the new destination Oracle database containing a get and put procedure.

Note: The put procedure is used to store the SQL statement result set in an array. The get procedure is used to retrieve the SQL statement result set from the array and execute it within the front end application. It is called inside the application after calling the stored procedure.

2. Replace all result sets, which you had previously commented out, with reference to the put procedure.

3. Migrate to the destination Oracle database.

Note: You must modify the front end application in order to use the get procedure to retrieve the result set.

The following exists after you implement the work around:

```
create procedure test
as
declare
  @x varchar(10)
select @x = "[Customer]"
result_set.put('select '' '|| test.x ||' '' from dual')
result_set.put('select * from ln_display where ptid >1')
```

Complete Oracle Package

The following lines are an example of an implementation of the result_set package.

```
PACKAGE Result_set IS
TYPE varchar2array IS table of varchar2(2000) index by binary_integer;
pointer integer :=1;
store_counter integer :=1;
rset_store varchar2array;
procedure put(data_to_save in varchar2);
procedure get (data_stored out varchar2);
END;
```

The PACKAGE BODY result_set is as follows:

```
procedure put(data_to_save in varchar2)
is
status integer :=0;
begin
  rset_store(store_counter) := data_to_save;
  store_counter := store_counter + 1;
  status := 1;
exception
when others then
  status := 0;
end put;
procedure get (data_stored in out varchar2)
```

```
is
status integer :=0;
begin
if pointer < store_counter then
data_stored := rset_store(pointer);
pointer := pointer + 1;
status := 1;
else
data_stored := '~';
status := 0;
end if;
exception
when others then
status := 0;
end get;
END;
```

How are result sets and dynasets returned to the calling program?

The Migration Workbench parser adds an extra argument of type REF CURSOR for result sets and dynasets. The REF CURSOR type is understood and can be manipulated by both PL/SQL and Oracle JDBC.

ODBC drivers support REF CURSORS. This means that the extra argument must be explicitly handled by the application and the client application code must be changed. However, some third-party vendors, such as Intersolv, supply ODBC drivers for Oracle that support REF CURSORS and can implicitly make use of REF CURSORS for using result sets/dynasets.

Therefore, no change is required in the client application code. The Oracle9i ODBC driver provides this ability also.

This is illustrated in the following examples of a Microsoft SQL Server or Sybase Adaptive Server stored procedure and its equivalent Oracle package and stored procedure as generated by the Migration Workbench parser.

Note: This example assumes that you already have set up an RDO connection.

Example 4–1 Microsoft SQL Server and Sybase Adaptive Server Stored Procedure

```
CREATE PROCEDURE byroyalty
AS
select au_id from titleauthor
GO
```

Example 4–2 Oracle9i Package and Stored Procedure WEAK REF CURSOR

The following is an example of the Oracle9i package and stored procedure WEAK REF CURSOR as produced by the Migration Workbench.

```
CREATE OR REPLACE ACKAGE "OMWB_EMULATION"."GLOBALPKG" AUTHID
CURRENT_USER AS
identity INTEGER;
trancount INTEGER;
TYPE RCT1 IS REF CURSOR; /*new weak cursor definition added*/
END globalPkg; /*Only one package required not one package per ref cursor type*/

TYPE RCT1 IS REF CURSOR RETURN RT1; END; PROCEDURE byroyalty( RC1 IN OUT
"OMWB_EMULATION"."GLOBALPKG".RCT1) AS
Sto0_selcnt INTEGER; Sto0_error INTEGER; Sto0_rowcnt INTEGER; Sto0_errmsg
VARCHAR2(255);
Sto0_sqlstatus INTEGER; BEGIN

OPEN RC1 FOR
SELECT au_id FROM titleauthor;

END byroyalty;
```

Example 4–3 Oracle9i Package and Stored Procedure STRONG REF CURSOR

The following is an example of the Oracle9i package and stored procedure STRONG REF CURSOR which the customer can use by altering the code manually.

```
PACKAGE BYROYALTYPkg AS
TYPE RT1 IS RECORD (

au_id titleauthor.au_id%TYPE
);

TYPE RCT1 IS REF CURSOR RETURN RT1; END; PROCEDURE byroyalty( RC1 IN OUT
byroyaltyPkg.RCT1) AS Sto0_selcnt INTEGER; Sto0_error INTEGER; Sto0_rowcnt
INTEGER; Sto0_errmsg VARCHAR2(255);
Sto0_sqlstatus INTEGER; BEGIN
```

```
OPEN RC1 FOR
SELECT au_id FROM titleauthor;

END byroyalty;
```

Example 4–4 Typical ODBC Code Used by Intersolv

The following example illustrates the typical ODBC code used by Intersolv to call the preceding Microsoft SQL Server or Sybase Adaptive Server stored procedure. This code also works for the preceding Oracle9i / Oracle8i package and stored procedure.

You must add error handling to a real application:

```
SQLPrepare(..., '{call byroyalty()}', ...)
SQLExecute()
SQLBindCol()
SQLFetch()
```

The following table describes the typical ODBC code used by Intersolv to call the stored procedure

Command...	Description
SQLPrepare(..., '{call byroyalty()}', ...)	Calls the ODBC SQL syntax used to execute stored procedures.
SQLExecute()executes	Executes the stored procedure.
SQLBindCol()assigns	Assigns storage for result column 1 in the result set (au_id).
SQLFetch()	Fetches the first record from the result set generated by the stored procedure.

The following examples illustrate how you call the preceding Microsoft SQL Server or Sybase Adaptive Server stored procedure with result sets / dynasets in Visual Basic using DAO and RDO on top of ODBC. This code works for Oracle8i packages and stored procedures if you use an Intersolv ODBC driver or the Oracle8i ODBC driver release 8.1.5.4.0 to understand Oracle REF CURSORS.

Example 4-5 DAO

The following examples illustrate how to call the Microsoft SQL Server or Sybase Adaptive Server stored procedure using DAO.

```
Private Sub Command2_Click()
Dim sSql As String
    sSql = "{call byroyalty()}"
    'In Oracle ODBC driver use refcursor argument to get result set
    Set rCustomers = dbsServer.OpenRecordset(sSql, dbOpenDynamic)
    Text4 = rCustomers.Fields(0) theend:
End Sub
```

Note: This example assumes that a DAO connection is set up.

Example 4-6 RDO

The following examples illustrate how to call the Microsoft SQL Server or Sybase Adaptive Server stored procedure using RDO.

```
Private Sub Command1_Click()
    StrSql = "{call byroyalty}"
    'in oracle odbc driver uses refcursor argument to get result set
    Set Ps = connx1.CreatePreparedStatement("PsTest", StrSql)
    Set Rs = Ps.OpenResultSet(rdOpenStatic)
    Text3 = Rs!au_id
    Rs.Close
End Sub
```

Can I use the Migration Workbench to migrate only stored procedures?

The Migration Workbench works first by capturing the source database. It loads the entire data dictionary of the source database into the Source Model and creates the Oracle Server equivalent called the Oracle Model. After the capture is complete, you can run the Migration Wizard. This forces you to create all users and their tables. After the users and tables are created, you can create the stored procedures.

You should capture the entire database. You can then use the Migration Wizard to migrate only the stored procedures by doing the following:

- In step 2, choose **Next** to create the users and tables. If you are recreating the users and tables, choose **Yes**.
- In step 3, choose **No** to migrate the table data to an Oracle database.

- In step 4, move all the schema objects to the Available list except for Packages and Procedures.

How are DDL commands handled?

The parser does not handle some DDL commands in stored procedures, for example the `DROP INDEX`, `CREATE VIEW`, `DROP VIEW`, `GRANT`, `DROP PROCEDURE`, and `CREATE INDEX` commands. You can use dynamic PL/SQL to emulate some of these commands.

Can you use DLL commands in an Oracle stored procedure?

Yes. You can use DLL commands by using the `dbms_sql` package.

Does the Migration Workbench support bitwise operations?

Yes. The Migration Workbench has limited support for bitwise operations. However, you might have to manually edit the stored procedures. Some PL/SQL functions were written to simulate `BITNAND` operations.

Can I parse a specific stored procedure?

Yes. You can do this by selecting a specific stored procedure and choosing the **Object > Parse** option within the Migration Workbench.

How are cross-database procedure calls converted?

Microsoft SQL Server and Sybase Adaptive Server databases are mapped to tablespaces instead of to users. Therefore, procedures in one database that call procedures in another database must be edited after migration.

For example, there are two databases called `HR` and `FINANCE`, owned by the user `SA`. There is a procedure called `get_sales_employees` in the `HR` database and a procedure called `get_quarter_bonuses` in the `FINANCE` database. If `get_quarter_bonuses` makes a call to `get_sales_employees`, the syntax is as follows:

```
sa..get_sales_employees
```


The Migration Workbench maps individual databases to separate tablespaces and creates both stored procedures (`get_sales_employees` and `get_quarter_bonuses`) under the database owner, which is often SA. Therefore, the previous statement is converted to the following:

```
hr.get_sales_employees
```

In the Oracle Server, the user called SA must own `get_sales_employees`.

See Also: For more information, see ["Parsing"](#) on page 4-8.

How does the Migration Workbench migrate Microsoft SQL Server outer joins to Oracle outer joins?

The Migration Workbench does not convert full outer joins to an Oracle database.

The Migration Workbench converts a SQL Server inner join to an Oracle join as illustrated in the following code example:

```
select * from a inner join b on a.col1=b.col2;
select * from a , b where a.col1=b.col2;
```

The Migration Workbench converts left joins and right joins in SQL Server to an Oracle database, for example:

```
select * from a left join b on a.col1=b.col2;
remember the table to the right of left join and add (+) to it's columns.
select * from a, b where a.col1=b.col2(+);
```

```
select * from a right join b on a.col1=b.col2;
remember the table to the left of right join and add (+) to it's columns.
select * from a, b where a.col1(+)=b.col2;
```

How can I convert full outer joins from a Microsoft SQL Server database to an Oracle database?

There are several ways to express a full outer join within an Oracle database. For example, in the following query the predicate `a.col1 (+) = b.col1 (+)` is similar to the way Oracle notates a full outer join although this predicate is not currently supported in an Oracle database:

```
select a.col2 acol2, a.col1 acol1, b.col1 bcol1, b.col2 bcol2
from a,b where a.col1 (+) = b.col1 (+);
```

The most efficient way of executing this query is to use a **UNION ALL** of a left outer join and a right outer join, with an additional predicate, for example:

```
select a.col2 acol2, a.col1 acol1, b.col1 bcol1, b.col2 bcol2
from a,b where a.col1=b.col1(+)
union all
select a.col2 acol2, a.col1 acol1, b.col1 bcol1, b.col2 bcol2
from a,b where a.col1(+)=b.col1 and a.col1 is null;
```

How do Microsoft SQL Server outer joins compare to ODBC?

The Microsoft SQL Server outer joins are supported by the Oracle ODBC driver. The following example illustrates how the Migration Workbench migrates Microsoft SQL Server outer joins to ODBC escape sequences. The Oracle ODBC driver release 9.0.1 or later supports ODBC escape sequences.

Example 4-7 Original SQL Server Statement

```
SELECT A.A,B.B
FROM A LEFT OUTER JOIN B ON A.A = B.B
WHERE ((A.C=3));
```

Example 4-8 Oracle ODBC Statement

```
SELECT A.A,B.B
FROM {oj} A LEFT OUTER JOIN B ON A.A = B.B }
WHERE ((A.C=3));
```

How does the Migration Workbench handle complex case statements?

The Migration Workbench converts **CASE** to **DECODE**. However, you must convert each condition in the case statement to a function call in decode. The Migration Workbench automatically converts the following syntax:

< <= = <!= != < !, AND, OR, NOT, IS NULL, LIKE

In complex case statements you might want to manually alter the PL/SQL code. To do this, you must first comment out the case statement in the source database procedure, then edit it within the Oracle database. The Migration Workbench does not convert commented code.

The following is an example of how to convert CASE to DECODE for syntax that the Migration Workbench does not convert automatically:

```

CASE N0 WHEN 1 THEN n1 ELSE n2 END)
  from emp
  becomes-----

      SELECT SUM(DECODE(N0, 1, n1 , n2))
      INTO GSGetAuftragnr.myvar FROM emp;
and
  select @myvar=sum(
CASE WHEN N0 = C0 AND N0 <0 THEN ((N0 - C0) / N0) * 100 ELSE 0 END)
  from emp
  becomes -----

      SELECT
      SUM(
      DECODE(1,
      DECODE(1,
      DECODE(greatest(N0,C0),N0,1,0)/*expr !<|= expr*/
      ,
      DECODE(N0,0,0,1)/*expr <|!= expr*/
      ,0)/*expr AND expr*/
      ,( ( N0 - C0 ) / N0 ) * 100,0))
      INTO
      GSGetAuftragnr.myvar FROM emp;

```

Global Variables

The following questions are about SQL Server and Sybase Adaptive Server global variables.

Does the Migration Workbench support global variables?

Yes. The Migration Workbench currently supports the following global variables:

- @@ERROR
- @@FETCH_STATUS
- @@IDENTITY
- @@ROWCOUNT
- @@SQLSTATUS

- @@TRANCOUNT
- @@VERSION

How does the Migration Workbench implement the migration of global variables?

The Migration Workbench creates equivalent functions in the destination Oracle database for global variables that are specific to stored procedures, for example the @@ROWCOUNT variable. The equivalent function is then called by the stored procedure.

Currently, the Migration Workbench includes the @@IDENTITY and @@TRANCOUNT variables into the `omwb_emulation` package. However, you can add additional global variables after migration. For variables such as @@TRANCOUNT and @@IDENTITY, the Migration Workbench does the following:

1. The Migration Workbench creates a new user called `omwb_emulation` with the password `oracle` in the destination Oracle database.
2. Inserts an invoker-rights package into the database within the new user schema.

Note: Unlike definer-rights routines, invoker-rights routines are not bound to a particular schema. A variety of users can run them. For more information, see the *PL/SQL User's Guide and Reference*.

3. Executes the database privilege `Grant execute to public` on the invoker-rights package. This allows any user within the destination Oracle database to query and update the defined variables.

How are IDENTITY columns mapped to an Oracle database?

A NUMBER column with an associated sequence and trigger is created in the destination Oracle database for each IDENTITY column in the Microsoft SQL Server or Sybase Adaptive Server database. Each time a row is inserted, the trigger queries the sequence for the next value and inserts that value into the IDENTITY column. Additionally, this value is inserted into the `omwb_emulation` user you created as part of the @@IDENTITY global variable. This allows the Migration Workbench to emulate the T/SQL @@IDENTITY global variable within the Oracle Model.

How does the Migration Workbench handle the @@SERVERNAME global variable?

The global variable @@SERVERNAME is treated as a normal variable within an Oracle Server. The Oracle Server equivalent is based on the name of the database where you are connected.

How does the Migration Workbench handle the @@SPID global variable?

The global variable @@SPID is treated as a normal variable within the Oracle Server. The Oracle equivalent might be based on the v\$SESSIONS table. Note that a user might not have a dedicated server process in the Oracle Server.

How does the Migration Workbench handle the @@TRANCOUNT global variable?

You can set the emulation of the @@TRANCOUNT global variable on or off by choosing the **Object > Parse** option within the Migration Workbench. When you have selected the parser option, each reference to the @@TRANCOUNT variable within the SQL Server stored procedure is replaced with a reference to the omwb_emulation.globalPkg.trancount packaged variable within the destination Oracle database. However, if you do not select the Emulate @@TRANCOUNT variable parser option, the @@TRANCOUNT variable is treated as a normal variable.

The following example compares the SQL Server statement with the equivalent Oracle Server statement when you select the Emulated @@TRANCOUNT variable option in the parse option.

SQL Server Statement

```
IF @@trancount > 100
```

Oracle Statement

```
IF omwb_emulation.globalPkg.trancount > 100
```

Note: You access the emulated variable through the typical user.package.variable format.

The SQL Server statements that affect the value of the @@TRANCOUNT variable, for example BEGIN TRANSACTION and COMMIT TRAN, are translated so they can update the omwb_emulation.globalPkg.trancount variable in the same way as within the source Transact SQL database. For example, in a Transact SQL database,

the BEGIN TRANSACTION statement increments the value of the global variable @@TRANCOUNT by 1. This statement is translated to achieve identical functional capabilities within the destination Oracle database, for example:

```
/* Emulating @@TRANCOUNT functionality in Oracle model */  
omwb_emulation.globalPkg.trancount:=omwb_emulation.globalPkg.trancount+1;
```

If the value of the @@TRANCOUNT variable is 1, the Transact SQL statement COMMIT TRANSACTION commits the work to the database and sets @@TRANCOUNT to 0.

If the value of the @@TRANCOUNT variable is greater than 1, the COMMIT TRANSACTION decrements the value of the @@TRANCOUNT variable only by 1.

This statement is translated to achieve identical functional capabilities within the destination Oracle database:

```
/* Emulating @@TRANCOUNT functionality in Oracle model */  
IF omwb_emulation.globalPkg.trancount = 1 THEN  
    COMMIT WORK;  
END IF;  
IF omwb_emulation.globalPkg.trancount > 0 THEN  
    omwb_emulation.globalPkg.trancount:=omwb_emulation.globalPkg.trancount-1;  
END IF;
```

How does the Migration Workbench handle the @@TRANSTATE global variable?

The @@TRANSTATE global variable is treated as a normal variable. There is no Oracle Server equivalent. Transactions in an Oracle Server are either complete or they fail. Errors in an Oracle Server are handled through exceptions and SQLCODE.

See Also: For more information, see the *PL/SQL User's Guide and Reference*.

Triggers and Rules

The following questions are about SQL Server and Sybase Adaptive Server triggers and rules.

Are there any issues with the use of deleted and inserted pseudo columns from Microsoft SQL Server?

The parser in the Microsoft SQL Server 7.0 plug-in incorrectly converts Microsoft SQL Server T-SQL stored procedures that contain the following:

```
SELECT @one=COL1, @two=COL2 FROM DELETED
```

The parser converts stored procedures to the following in PL/SQL:

```
SELECT :OLD.COL1, :OLD.COL2
      INTO one, two FROM DUAL;

SELECT @one=COL1, @two=COL2 FROM DELETED, fred
SELECT COL1, COL2
      INTO one, two FROM fred;
Sto0_rowcnt := SQL%ROWCOUNT;
```

You must manually correct PL/SQL stored procedures that are converted in this way. Rewrite the code as follows:

```
SELECT :OLD.COL1, :OLD.COL2
      INTO one, two FROM fred;
Sto0_rowcnt := SQL%ROWCOUNT;
```

See Also: For more information, see ["Parsing"](#) on page 4-8.

How are rules handled?

Rules are migrated as table check constraints. This means that every table column that has a rule associated with it has a check constraint in the destination Oracle database. Patterns, such as LIKE[a-y], are translated using TRANSLATE.

How are rules with the getDate() function converted?

Rules are converted to equivalent Oracle Server check constraints. The getDate() function is semantically equivalent to SYSDATE in an Oracle Server. However, Oracle Servers do not support the use of SYSDATE in check constraint definitions. Therefore, a database trigger must be created in the Oracle Server to implement rules that use the getDate() function.

See Also: For more information, see Rules in the *Oracle Migration Workbench Reference Guide for Microsoft SQL Server and Sybase Adaptive Server Migrations*.

MySQL Server Questions

This chapter contains frequently asked questions about using the Oracle Migration Workbench to migrate from a MySQL Server 3.22 or 3.23 database to an Oracle Server. It contains the following sections:

- [Data Migration](#)
- [Migration Error Messages](#)

Data Migration

This section contains a MySQL data migration question.

Are there any guidelines for how long it takes to convert high volumes of data from MySQL to an Oracle Server using the Migration Workbench?

No. Performance tests have not been carried out on the Migration Workbench data migration process. However, it is known that the setup affects performance. For example, if the Oracle Server and MySQL are not located on the same server, you must take the network bandwidth into account. Presently, data migration involves reading the data from the source MySQL database through a JDBC bridge and writing the data to the Oracle database through JDBC. It is a straightforward data pump. However, the Migration Workbench takes advantage of array inserts and 'commit points' to improve performance. You can use the `mysqldump` scripts in conjunction with `SQL*Loader` to provide offline data loading for non-binary data.

Migration Error Messages

This section contains a MySQL error message question.

Why do I receive the "Communication link failure: bad handshake" error message when migrating from MySQL?

You may receive this error message during Step 1: Source Database Details of the Capture Wizard. To resolve this error message, verify the following conditions:

- The `root` user exists and has sufficient privileges on the MySQL server.
- The password you are using for the `root` user is correct.
- The password is exactly eight characters in length. In some versions of MySQL the password for the `root` user must be exactly eight characters to allow a connection through a third party product using JDBC or ODBC drivers. This is true even if a password that is not eight characters long allows you to connect through MySQL client programs.
- The server name, system (host) that you are connecting from has access rights to the MySQL user table.

List of Questions

This appendix lists the questions contained in the chapters of this FAQ.

General Migration Workbench Questions

What release of the Oracle Server does the Migration Workbench support?

What platforms does Oracle Migration Workbench run on?

Does the Migration Workbench support UNIX?

Can I download the Migration Workbench from the Web?

Why am I not able to access Oracle Technology Network (OTN)?

What should I do if I cannot download the Migration Workbench from Oracle Technology Network (OTN)?

Why do I receive a blank DOS window or one of the following error messages while launching the Migration Workbench from Windows 98?

Where can I find out more about Oracle Migration Workbench error messages?

What is the difference between Oracle Transparent Gateway (OTG) and the Migration Workbench?

Does the Migration Workbench provide NLS support?

Does the Migration Workbench support object names with spaces in their names?

Does the Migration Workbench use strong REF CURSORS?

Can I install the Migration Workbench into an Oracle home directory that contains a release of the Oracle Server earlier than Oracle9i?

How do I re-create an Oracle Migration Workbench Repository?

How do I create a tnsnames.ora file?

What should I do if I cannot launch the Oracle Migration Workbench? I attempt to do so but nothing happens.

I attempt to launch the Oracle Migration Workbench and the splash screen appears then quickly disappears. Why does this happen?

Can I migrate single table data?

Can I change the data type of a single column?

Are constraints disabled when loading the data?

How are duplicate object names handled?

What are the object naming guidelines for the Migration Workbench?

How does the Migration Workbench handle reserved words?

How does the Migration Workbench migrate datatypes that the Oracle Server does not support?

What do the following error messages mean?

How are NCHAR, NVARCHAR, NTEXT, or MEMO data types mapped within the Oracle Server?

Does the Migration Workbench use SQL*Loader to control files for loading data?

Can I modify the SQL*Loader control files for loading the data?

Does the data migration process slow down dramatically when migrating tables that contain large volumes of binary data, such as images, binary data, or OLE objects?

What should I do if I receive the error "ORA-22866: default character set is of varying width" when migrating to a destination Oracle database created with the UTF8 character set?

What is the meaning of the error message "ORA-01426: numeric overflow" in the Log window?

What does the error "Hostdef does not exist" mean?

Why do I receive the "ORA-02298: parent key not found" error message?

How do client applications access the data that has been migrated from my source database to an Oracle database?

Does the Oracle Server have implicit date conversion from a character string?

Can I migrate a single schema object?

Can I rename a schema object?

How do I manage case-sensitivity queries?

Can the Migration Workbench migrate multiple databases?

Does the Migration Workbench allow multi-character field delimiters?

How do I modify the storage options for a table?

Do I need to edit table defaults?

How does the Oracle Server handle temporary tables?

How do I get help on Migration Workbench issues?

How do I know what error messages occurred when running the Migration Wizard?

All error messages are logged to the Log window and the columns of the Log window can be sorted by clicking on the column header. During the creation phase of the migration, the Migration Wizard creates three log files that record all successful CREATE statements and ALTER statements, the equivalent DROP statements and ALTER statements, and all SQL statements that the Migration Workbench was unable to execute. The location of these log files is governed by the log file directory setting in the Logging page of the Options dialog box. The default location is %ORACLE_HOME%\omwb\log. If the Migration Workbench fails to respond, check the contents of the %ORACLE_HOME%\Omwb\log\Error.log file to see if there are any error messages recorded. The contents of this file can be sent to the Migration Workbench development team at infomwb_ww@oracle.com. You can also generate reports from the Migration Workbench. This provides you with a formatted list of the error messages.

Informix Dynamic Server Questions

Are there any schema migration limitations for Informix Dynamic Server?

What do I use to connect to an Informix Dynamic Server?

Why do I have problems migrating the Defaults schema when I use the default repository?

What must I do when the stored procedure parameters that are declared using the LIKE syntax are incorrectly parsed to type CLOB?

Why does the stored procedure parser fail in the Migration Workbench?

Does the Informix Dynamic Server Stored Procedures Parser have any limitations?

Does the Migration Workbench fully support the migration of LOB data using SQL*Loader scripts?

Microsoft Access Questions

Does the Migration Workbench support Microsoft Access 2000 migration to Oracle?

How do I migrate a Microsoft Access database to an Oracle database?

What software do I need to migrate a Microsoft Access 97 database to an Oracle database?

What software do I need to migrate a Microsoft Access 2000 database to an Oracle database?

What happens if the following message appears when you log into the Oracle Migration Workbench: "There is no Microsoft Access ODBC driver software installed on this system. Please install the software before using the Microsoft Access plugin."

Why does the Oracle Model hang when I map foreign keys?

What should I do if I receive the "ORA-00001: Unique constraint violated message displayed in error.log or as an alert" error message?

What if I receive one of the following messages while attempting to launch the Oracle Migration Workbench Exporter for Microsoft Access:

Why does the Migration Workbench not handle validation rules with the following syntax:

Can I migrate my Microsoft Access security settings to an Oracle Server?

How do I avoid erroneous relations within Microsoft Access 95?

Can I migrate a replica Microsoft Access database?

Can I use the Migration Workbench to migrate multiple Microsoft Access databases to a single Oracle user?

Can my hyperlinks work after I migrate my database to an Oracle Server?

What does the message "No primary key defined on table name; you are unable to update records after migration." mean?

What should I do if I encounter the error message "JET/DAO Error 3050: Couldn't lock file"?

Why must I define an Oracle ODBC data source when migrating from Microsoft Access?

Why must I modify the Microsoft Access database?

Does the Migration Workbench support offline data loading in Microsoft Access?

How do I migrate the Microsoft Access Memo columns in tables?

What should I do if the following error is reported during data load from Microsoft Access to Oracle: "EXCEPTION: LoadTableData.run() : [Microsoft][ODBC Microsoft Access Driver] '(unknown)' is not a valid path. Make sure that the path name is spelled correctly and that you are connected to the server on which the file resides"?

Microsoft SQL Server & Sybase Adaptive Server Questions

Does the Migration Workbench support Microsoft SQL Server 6.0? If not, are there any plans to support this version of Microsoft SQL Server?

Can I migrate from Microsoft SQL Server or Sybase Adaptive Server to an Oracle7 database using the Migration Workbench?

What is the meaning of the "Owner name is NULL. Check the MASTER database is correctly recovered" message?

Why must I define a SQL Server or Sybase System 11 ODBC data source?

Why does a SELECT statement on a temporary table fail when I have migrated a stored procedure to a destination Oracle database?

Can you migrate roles and privileges using the Migration Workbench?

How does the Migration Workbench map the Microsoft SQL Server database?

How are Microsoft SQL Server and Sybase Adaptive Server user-defined types handled?

What happens if I receive the following error message: "Failed to load Source Model [INTERSOLV] [IDBC SQL Server] [SQL Server] WARNING! Some characters could not be converted into client's character set. Unconverted bytes were changed to question marks (?)" ?

Does the Migration Workbench support offline data loading in Microsoft SQL Server or Sybase Adaptive Server data migration?

How do I run BCP from a client system?

Are there any guidelines for how long it takes to convert high volumes of data from Microsoft SQL Server or Sybase Adaptive Server to an Oracle Server using the Migration Workbench?

How are TIMESTAMP data-types migrated?

What is the Microsoft SQL Server or Sybase Adaptive Server DATETIME data type mapped to within an Oracle database?

How does the Migration Workbench handle Microsoft SQL Server 7.0 VARCHAR data types greater than 4000?

What is the equivalent of the Microsoft SQL Server or Sybase Adaptive Server SYSOBJECTS table in an Oracle database?

Can I edit a stored procedure, trigger, view, or check constraint in the Source Model?

How are Microsoft SQL Server and Sybase Adaptive Server EXEC statements parsed?

Can I use double brackets (()) in T/SQL statements?

Does the Migration Workbench parser support cast operations?

Can I use the LIKE clause in SQL statements?

Can I migrate subqueries in the SELECT clause of Microsoft SQL Server and Sybase Adaptive Server?

Does an Oracle Server support full UNIX-style regular expression handling?

Are DUMP TRANSACTION and DROP INDEX SQL commands supported?

How do I avoid problems with cyclic procedures?

Does the Migration Workbench support the conversion of T/SQL stored procedures that return multiple result sets?

How are result sets and dynasets returned to the calling program?

Can I use the Migration Workbench to migrate only stored procedures?

How are DDL commands handled?

Can you use DLL commands in an Oracle stored procedure?

Does the Migration Workbench support bitwise operations?

Can I parse a specific stored procedure?

How are cross-database procedure calls converted?

How does the Migration Workbench migrate Microsoft SQL Server outer joins to Oracle outer joins?

How can I convert full outer joins from a Microsoft SQL Server database to an Oracle database?

How do Microsoft SQL Server outer joins compare to ODBC?

How does the Migration Workbench handle complex case statements?

Does the Migration Workbench support global variables?

How does the Migration Workbench implement the migration of global variables?

How are IDENTITY columns mapped to an Oracle database?

How does the Migration Workbench handle the @@SERVERNAME global variable?

How does the Migration Workbench handle the @@SPID global variable?

How does the Migration Workbench handle the @@TRANCOUNT global variable?

How does the Migration Workbench handle the @@TRANSTATE global variable?

Are there any issues with the use of deleted and inserted pseudo columns from Microsoft SQL Server?

How are rules handled?

How are rules with the getDate() function converted?

MySQL Questions

Are there any guidelines for how long it takes to convert high volumes of data from MySQL to an Oracle Server using the Migration Workbench?

Why do I receive the "Communication link failure: bad handshake" error message when migrating from MySQL?

Index

A

accessing
 client applications, 1-12

B

BCP
 generating
 SQL*Loader scripts, 4-7
bitwise operations, 4-18

C

calling
 cross-database procedures, 4-18
case-sensitivity
 schema objects, 1-13
cast operations
 support for, 4-9
character strings
 implicit date conversion, 1-13
client applications
 accessing, 1-12
comparing
 Oracle Transparent Gateway, 1-4
complex case statements, 4-20
configuration, 1-5
configuration file, 1-6
creating
 repository, 1-6
cross-database procedure calls, 4-18
customizing
 storage options, 1-14

cyclic procedures
 avoiding, 4-11

D

data loading
 constraints, 1-7
 controlling files, 1-10
data types
 mapping, 1-8
 modifying, 1-7
 modifying DATETIME, 4-8
 modifying TIMESTAMP, 4-8
data types, modifying VARCHAR, 4-8
databases
 Microsoft SQL Server, 4-4
date conversion
 for character strings, 1-13
DDL commands, 4-18
DLL statements
 stored procedures, 4-18
double brackets
 using, 4-9
downloading, 1-3
dynasets, 4-14

E

editing
 schema objects, 4-9
 table defaults, 1-14
error messages, 1-11, 3-10
 creating Oracle Model, 1-9
 foreign keys, 1-12

- hostdef, 1-12
- loading source database, 1-9
- logging, 1-16
- Microsoft Access ODBC driver, 3-4
- more information, 1-4
- Sybase ODBC translator, 4-4

exporter

- for Microsoft Access, 3-2

G

global variables, 4-21

- @SERVERNAME, 4-23
- @SPID, 4-23
- @TRANSTATE, 4-24

I

IDENTITY columns, 4-22

incomplete

- source database, 1-10

incomplete repository, 1-9

installation, 1-5

installing

- Migration Workbench, 1-6

J

Java Runtime Environment

- default location, 1-7

L

logging

- error messages, 1-16

M

memo columns

- tables, 3-12

Microsoft Access 2000

- migrating, 3-2

Microsoft SQL Server 6.0

- support for, 4-2

migrating

- multiple databases, 1-14

modifying

- Microsoft Access, 3-11

multi-character field delimiters, 1-14

multiple databases, 1-14

O

object names, 1-5

- duplicates, 1-8

ODBC

- comparing to outer joints, 4-20

offline data loading, 4-5

- SQL*Loader, 3-11

operations

- bitwise, 4-18

Oracle Transparent Gateway

- comparing, 1-4

Oracle7 databases, 4-2

Oracle8

- support, 1-2

ordering, 1-3

outer joints

- comparing, 4-20
- converting, 4-19
- migrating, 4-19

P

parsing

- DUMP commands, 4-10
- EXEC statements, 4-9
- expression handling, 4-10
- LOAD commands, 4-10
- stored procedures, 4-18
- SYSOBJECTS, 4-9

performance tests, 4-7

platforms, 1-2

privileges, 4-3

pseudo columns, 4-24

purchasing, 1-3

R

re-creating

- repository, 1-6

REF CURSOR

- weak, 1-5

relations

- within Microsoft Access 95, 3-8

renaming

- schema objects, 1-13

replica databases, 3-9

repository

- creating, 1-6

- incomplete, 1-9

- re-creating, 1-6

reserved words, 1-8

result sets, 4-14

roles, 4-3

rules

- getDate functions, 4-25

- handling, 4-25

S

schema objects

- case sensitivity, 1-13

- editing, 4-9

- migrating, 1-13

- renaming, 1-13

scripts

- Microsoft Access, 3-11

- SQL*Loader, 4-5, 4-7

source database

- incomplete, 1-10

SQL statements

- LIKE clause, 4-10

SQL*Loader

- loading data, 1-10, 1-11

SQL*Loader scripts, 4-5

- BCP, 4-7

stored procedures

- DLL statements, 4-18

- migrating, 4-17

- multiple result sets, 4-12

- parsing, 4-18

support, 1-15

- for Oracle8, 1-2

- for Unix, 1-2

- help, 1-15

T

table defaults

- editing, 1-14

tables

- customizing options, 1-14

- memo columns in Microsoft Access, 3-12

- Microsoft SQL Server, 4-4

testing

- performance, 4-7

tnsnames.ora file, 1-6

- Oracle home instances, 1-7

U

Unix

- support, 1-2

user-defined types

- handling, 4-4

using

- LIKE clause, 4-10

