CA ERwin® Data Modeler Workgroup Edition

Implementation Guide

Release 8.2.1

CA Technologies Product References

This document references the following CA Technologies products:

- CA ERwin® Data Modeler Workgroup Edition (CA ERwin DM WE)
- CA ERwin® Data Modeler Navigator Edition (CA ERwin DM NE)

Contact CA Technologies

Understanding your Support

Review support maintenance programs and offerings.

Registering for Support

Access the CA Support online registration site to register for product support.

Accessing Technical Support

For your convenience, CA Technologies provides easy access to "One Stop" support for all editions of CA ERwin Data Modeler, and includes the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

For information about other Home Office, Small Business, and Enterprise CA Technologies products, visit http://ca.com/support.

Provide Feedback

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA Technologies product documentation, complete our short customer survey, which is also available on the CA Support website, found at http://ca.com/docs.

CA ERwin Data Modeler News and Events

Visit www.erwin.com to get up-to-date news, announcements, and events. View video demos and read up on customer success stories and articles by industry experts.

Documentation Changes

The following documentation updates have been made since the last release (r8.1) of this documentation:

- Prepare for the Installation (see page 15)—Information was added to advise you to see the *CA ERwin Data Modeler Installation Guide* for more information about installing CA ERwin Data Modeler. The Installation Guide is a new guide added for this release, which provides you with all the information necessary for installing and licensing CA ERwin Data Modeler. Information about how the r8 upgrade process affects models in the mart was also added.
- The topics *Install CA ERwin Data Modeler Workgroup Edition* and *Get Licenses* were removed from this guide. All specific information about installing and licensing this product is now in the Installation Guide.

Contents

Chapter 1: Modeling in the Multiuser Environment

CA ERwin Data Modeler Workgroup Edition coordinates the development and management of data models created with CA ERwin Data Modeler.

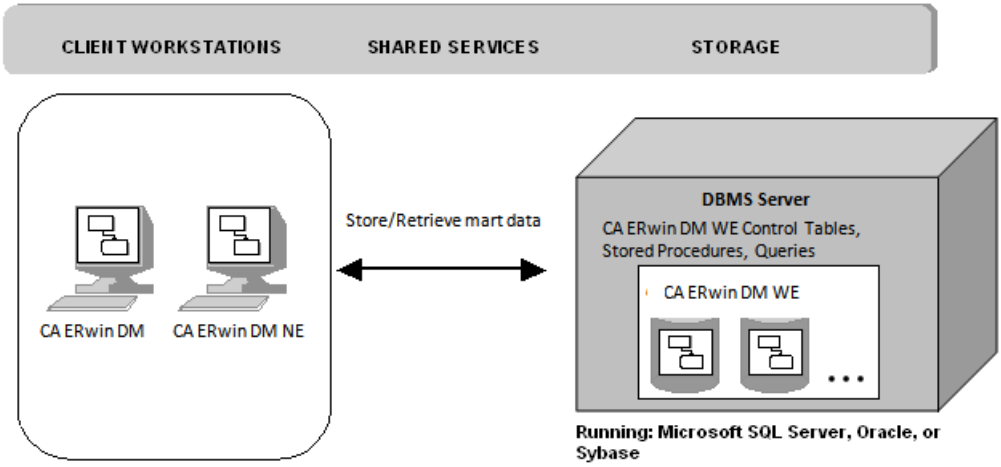
This section contains the following topics:

CA ERwin Data Modeler Workgroup Edition

CA ERwin Data Modeler Workgroup Edition provides a multiuser modeling environment that makes coordinated, large-scale modeling possible. It enables collaboration among project managers, data modeling team members, and standards administrators by providing workgroup modeling services, including conflict resolution, versioning, security, and standards management. You can coordinate the efforts of model teams to document existing systems, create new eBusiness systems, and drive data standardization. Model sharing encourages teamwork, so modelers can work together more efficiently to optimize model-based development. Your modelers get more work done in less time with better results.

Workgroup modeling operates in a client/server environment so processing is shared between the user client workstation and the server where the models are stored. Users access the stored models through a client, such as CA ERwin Data Modeler. Models are held in a platform and network independent database or mart that resides on a central server, using Microsoft SQL Server, Oracle, or Sybase as the host database.

The following diagram shows a typical CA ERwin Data Modeler Workgroup Edition environment:



Workgroup modeling features are provided to help control updates to models when you are opening, closing, and saving models. The administrator installs the program and initializes the database on the DBMS server. The administrator also has the responsibility of setting up the library structure in which models are organized, and assigning security profiles to users. Administration overhead is relatively low and proportional to the number and activity level of the users. After a model is saved to the database, control of who can work on the model and how changes are saved to the model is handled by security, making it possible for workgroups to work on large models without confusion. CA ERwin Data Modeler Workgroup Edition supports many security and administrative features related to its multiuser capabilities and the client/server environment in which it operates. For this reason, routine management is usually performed by a dedicated administrator. Administrators can find detailed information regarding their tasks and responsibilities in the *Administration Guide*.

You can also use CA ERwin Data Modeler Navigator Edition as a client. Read-only access is permitted to data models, so your workgroup can use this client to share information with others without the risk that unauthorized changes might be saved to the mart. Additionally, you can connect to the mart through the Naming Standards Editor or the Datatype Standards Editor.

Model Life Cycle Frameworks

Use one of the following model life cycle frameworks in your organization:

Model-Driven Development

Changes to the schema are made to the model first and then forward-engineered.

System-Driven Models

Changes are made directly to the schema and the schema is reverse-engineered into the model to reflect the changes.

Informational Models

Contains logical-only models, enterprise-wide models, or standards and sample models.

Each type of framework has different considerations that you must think about when configuring your database and developing its supporting policies. You are not required to choose a particular framework, however, it helps to know your development process before building a library structure.

Model-Driven Development Framework

In the model-driven framework, the model is always the source of all changes. You create a new database schema by forward engineering the model. The life cycle of a model in the model-driven Development framework can follow a path like this:

- Create the library structure (for example, Development, Test, and Production) and populate them with CA ERwin Data Modeler templates.
- Create the logical model in a development library.
- Promote the model to the test library when it is ready.
- Generate the schema from the test library.
- Modify the test model as required and synchronize it to the schema.
- Promote the model to the production library when it is ready.
- Publish the refreshed production model.
- Update the enterprise-wide model, if necessary.
- Incorporate changes into the development model for further changes, and repeat the process.

System-Driven Model Framework

In the system-driven framework, there is an established information system from which you can reverse engineer database tables. The life cycle of a model in the system-driven Development framework can follow a path like this:

- Create the library structure (for example, Reverse Eng, Test, Production). You should not require CA ERwin Data Modeler templates because you do not create models from scratch.
- Reverse engineer the model from the information system into the designated library.
- Enhance the model with logical information and input from analysts.
- Update the model to reflect changes in the physical schema.
- Create a version of the model.
- Synchronize the schema and the model using Complete Compare.
- Publish the model.
- Repeat the last three steps as the system is modified.

Informational Model Framework

In the Informational Model framework, CA ERwin Data Modeler Workgroup Edition contains logical-only models, enterprise-wide models, or standards and sample models. There is no forward engineering with the intent of using the schema. The life cycle of a model in the Informational Model framework can follow a path like this:

- Create the library structure and populate them with templates.
- Develop the initial model. Use reverse engineering and model new components as required.
- Publish the initial model to the appropriate parties for modification and refinement.
- Get approval, and then version the model.
- Publish the approved model.
- Update and publish models as the enterprise model evolves.

Chapter 2: Installation and Configuration

This section contains the following topics:

Prepare for the Installation

Review the hardware and software requirements and the list of supported databases in the *CA ERwin Data Modeler Release Notes* before you install the program. You must verify that your system requirements meet at least the minimum specified requirements, and that you have verified and prepared your DBMS environment for installation.

Review the installation and licensing information in the *CA ERwin Data Modeler Installation Guide*. All of the information necessary to properly install and license this product is in this guide.

Before you can install CA ERwin Data Modeler Workgroup Edition in your DBMS environment you must do the following:

1. Create the database on your DBMS server.
2. Install or upgrade your license permitting access to and initialization of the mart.
3. Create the mart or upgrade your current version of software when a newer version is available.

Note: To create, update, or delete a mart in CA ERwin Data Modeler Workgroup Edition, you must have the following database rights:

- For Microsoft SQL Server 2000/2005/2008, you must be the database owner (dbo).

  Notes:

  - The dbo is no longer required to have the sysadmin role.
  - The SQL Server 2005/2008 TRUSTWORTHY database property is no longer required for the mart. After you install this product, you can optionally reset this property to OFF.
- For Oracle, you must be the database schema owner and have the DBA role.

Specify Use of Foreign Characters With Microsoft SQL Server 2000

For Microsoft SQL Server 2000, select specific settings in the Client Network Utility to have certain foreign language characters in your models recognized.

Follow these steps:

1. Click Programs, Microsoft SQL Server, Client Network Utility on the Start menu.

   The SQL Server Client Network Utility dialog opens.

2. Select the following check boxes on the DB-Library Options tab:

   - Automatic ANSI to OEM conversion
   - Use international settings

3. Click OK.

   Your configuration is set to recognize foreign language characters in your models.

Specify Use of Foreign Characters With Microsoft SQL Server 2005

For Microsoft SQL Server 2005/2008, modify your registry settings to have certain foreign language characters in your models recognized.

Follow these steps:

1. Click Run on the Start menu.
2. Enter *regedit*.

   The Registry Editor opens.

3. Verify or add the following registry entry:

   [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\Client\DB-Lib]

   "AutoAnsiToOem"="ON"

   "UseIntlSettings"="ON"

4. Click File, Exit.

   Your configuration is set to recognize foreign language characters in your models.

How to Prepare Your DBMS Environment

Perform the following setup tasks in your DBMS environment before you can install and use the software:

1. Install the DBMS on the server where you plan to store the mart.

   For more information about memory and disk space requirements, see the system requirements for each DBMS (Microsoft SQL Server, Sybase, and Oracle).

2. Create the mart.

   Use the DBMS features to create or identify the required storage objects and the mart. The specific requirements vary depending on your DBMS type.

3. Create a user profile for each CA ERwin Data Modeler Workgroup Edition user.

   Use the DBMS features to create a user profile for each user in the database where the mart is stored.

   Note: Users must have Windows authentication enabled on databases that support Windows authentication. In addition, users must have valid user accounts and permissions to access the database where the mart is stored. For more information, see your DBMS documentation.

4. Install the DBMS client software on each client workstation that has to access the mart.

   To connect a client workstation to the DBMS server, install the appropriate client connection software (for example, SQL*Net) on each workstation. The connection software you use depends on your DBMS type. Install the client so that any user with a mart user account and the appropriate DBMS client connection software can connect to the mart.

5. Verify the client/server connection.

   Use a test command (for example, *ping*) to verify the client/server connection between each client workstation and the DBMS server. Alternatively, use another application on a client workstation (for example, ISQL/W, SQLPLUS, or PowerBuilder) to verify the connection. For more information about testing the client/server connection, see your DBMS documentation.

Tasks to Create a Microsoft SQL Server DBMS

The following tasks must be performed by the DBA and system administrator responsible for installing CA ERwin Data Modeler Workgroup Edition on a Microsoft SQL Server DBMS:

1. Use a graphical user interface (GUI) tool, ISQL (all versions), SQL Administrator, or Enterprise Manager to create the CA ERwin Data Modeler Workgroup Edition database. If a GUI tool is not available, you can use ISQL to type in the appropriate commands manually.

   Your database should meet the following criteria:

   - The initial size of the database file should be set to 60 MB.
   - The initial size of the transaction log file should be set to 50 MB.
   - Set the maximum file size to unrestricted file growth for both files (recommended, but not required).
   - Increase the Set Auto grow file by 10 percent (recommended, but not required).

   The new database is *owned* by the user who created it.

   Set the Truncate Log on Checkpoint option and have the server generate checkpoints frequently. By selecting this option, the log is emptied periodically and should not fill up and cause rollbacks.

   Note: For best performance ensure that separate devices are used to store the data and the transaction log.

2. Add users.

   For users to access the mart you must either add existing logins as users to the database or create new logins and add them as users of the database. A login is an entity permitted to connect to the database server; a user gives a login the right to use a particular database. If you use Windows user names and passwords to secure database access (Windows Authentication), you must add the Windows users to your DBMS.

3. Verify tempdb size.

   Significant temporary space is required for installation and use. The temporary segments need at least 16 MB of available space. You should also increase available space as the number of concurrent users increase.

Microsoft SQL Server Connections

If you are using a Microsoft SQL Server as your host DBMS, CA ERwin Data Modeler Workgroup Edition uses DB_LIBRARY to connect.

Microsoft SQL Server 2005 Permissions

For SQL Server 2000, you only need the public permission assigned to save to the mart. However, when the repository is on a SQL Server 2005 instance, you must have the *bulkadmin* permission designated too. The ability to do bulk inserts (which the public permission permitted previously) is no longer part of the public permission. As the administrator, you explicitly define this permission. If you do not define, when you create a mart using a SQL Server 2005 database, and save it, an error appears.

Tasks to Create a Sybase DBMS

The DBA and the system administrator responsible for installing the software on a Sybase database management system perform the following tasks:

1. Use a graphical user interface (GUI) tool, ISQL (all versions), SQL Administrator, Sybase Central Java, or Enterprise Manager to create the mart database. If a GUI tool is not available, use ISQL to type the appropriate commands manually.

   Your database must meet the following criteria:

   - The size of the data device you create determines the size of the database. The minimum database size is 32 MB so there has to be at least one device that is 32 MB. Create the data device on a different disk (and disk controller) than the transaction log.
   - For optimum performance, verify that separate devices are used to store the data and the transaction log. For example, you can increase performance by creating a 50-MB data device and a 25-MB log device. A minimum of 40 MB of disk space (data and log) is required.

2. Add users.

   For users to access the mart, either add existing logins as users to the database, or create logins and add them as users. A login is an entity permitted to connect to the database server; a user gives a login the right to use a particular database. If you use Windows user names and passwords to secure database access (Windows Authentication), add the Windows users to your DBMS.

3. Verify the Stored Procedure Cache.

   Set the Stored Procedure Cache size to at least 8 MB. Setting it higher improves performance, especially when many users are accessing the server concurrently. Setting it lower results in fatal errors and rollbacks when the Stored Procedure Cache size is exceeded.

   Note: The installation creates more than 100 stored procedures. The client invokes these stored procedures to control changes to the data in the database.

4. Verify tempdb size.

   Significant temporary space is required for installation and use. The temporary segments need at least 16 MB of available space. Increase the available space as the number of concurrent users increase.

5. Verify the memory allocated to the database server.

   Allocate at least 32 MB of RAM to the database server. The amount of RAM allocated is ideally half of the available RAM on the server.

Sybase Connections

If you are using Sybase as your host DBMS, CA ERwin Data Modeler Workgroup Edition uses CT_LIBRARY to connect.

Post Database Creation Tasks for Microsoft SQL Server and Sybase

After you create the database, run the Administrative Setup program located on the CD. Provide the installation directory. By default this directory is C:\Program Files\CA\ERwin Data Modeler r8, but the path may have been changed during installation.

Note: Do not attempt to run the Setup program unless you are certain that the computer you are using can connect to the target server. For more information about how to determine if your installation computer can connect to the target server, see Microsoft SQL Server Connections (see page 18) or Sybase Connections (see page 20).

When you install the software, the Microsoft SQL installer can be ‘owner,’ ‘user with DB owner privileges,’ or ‘user with alias as DB owner privileges.’ The Sybase installer can be ‘owner,’ ‘user with sa_role,’ or ‘user with alias as DB owner.’

Transact-SQL Commands

If a graphical DBMS access tool is not available, you can use Transact-SQL commands through ISQL.

Example: Create a device using the Transact-SQL DISK INIT command through ISQL

```
DISK INIT NAME = 'mmdata',       /* The logical name. */
PHYSNAME = 'C:\SQL\DATA\mmdata.dat',   /* The physical name. */
VDEVNO = 1<= virtual_device_number => 255
/* System dependent. */
  SIZE = number_of_2K_blocks     /* 1024 here is 2MB!!! */
  [, VSTART = virtual_address,      /* Optional */
CNTRLTYPE = controller_number]              /* Optional */
```

Example: Create a database using the Transact-SQL CREATE DATABASE command through ISQL

```
CREATE DATABASE mmmaster
  [ON {DEFAULT | database_device} [= size_in_megabytes] /* The device created in #1. */
  [, database_device [= size_in_megabytes]]...]   /* A database can span devices. */
  [LOG ON database device [= size_in_megabytes>]   /* Separate log device. */
  [, database device [= size_in_megabytes]]...] /* A transaction log can span devices. */
```

Example: Add logins to the database with the *sp_addlogin* and *sp_adduser* commands using Transact-SQL through ISQL

```
sp_addlogin login_id [, passwd [, defdb [, deflanguage]]]

sp_adduser login_id [, username [, grpname]]
```

After you execute these commands, the DBA can alias an existing login as the Database Owner (dbo) or change the dbo to an existing login using *sp_changedbowner*. Use ISQL to execute the following:

```
sp_changedbowner login_id [,true]
```

Tasks to Create an Oracle DBMS

The DBA and the system administrator responsible for installing the software on an Oracle database management system performs these tasks.

Use graphical tools or SQL *Plus (all versions), SQL*DBA, or the Oracle Enterprise Management Console to perform these tasks. Examples of SQL commands are included where appropriate. Data file paths, data file sizes, role names, and user names are included for example only.

1. Check SYSTEM tablespace.

   The installation creates several stored procedures. All triggers, stored procedures, and packages are kept in the Oracle SYSTEM tablespace. The standard size of the SYSTEM tablespace assumes that you are not using procedural options, so the SYSTEM tablespace often needs to be expanded. If other Oracle applications are not using procedural code, then the SYSTEM tablespace should be expanded to 32 MB. If other Oracle applications also use procedural code, expand the SYSTEM tablespace to at least 32 MB.

2. Check Rollback Segment tablespace.

   If your instance uses UNDO tablespace, do not create rollback segments.

   Significant rollback space is required for installation and use. The rollback segments should be in their own separate tablespace and each have at least 16 MB of available space. There should be one rollback segment for every four concurrent users, with a maximum of 50 rollback segments. The available space should scale upward with increasing numbers of rollback segments. Finally, the rollback segment optimal parameter should be set to control rollback segment growth and space consumption.

   Note: For Steps 3, 4, and 5, use Dictionary-managed tablespaces.

3. Create a data tablespace of at least 32 MB.

   For example:

   ```
   CREATE TABLESPACE MyMart

   DATAFILE '/db01/oracle/rdbms9i/data/mymart.ora' SIZE 100M;

   Or For Locally Managed extents:

   CREATE TABLESPACE Mymart

   DATAFILE '/db01/oracle/rdbms9i/data/mymart.ora' SIZE 100M

   EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;
   ```

4. Create a index tablespace of at least 32 MB.

   For example:

   CREATE TABLESPACE MMARTINDEX

   DATAFILE '/db02/oracle/rdbms9i/data/mmartindex.ora' SIZE 75M;

   Or For Locally Managed extents:

   CREATE TABLESPACE MMARTINDEX

   DATAFILE '/db02/oracle/rdbms9i/data/mmartindex.ora' SIZE 75M

   EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;

5. Create a temporary tablespace.

   For example:

   CREATE TEMPORARY TABLESPACE MMTEMP TEMPFILE '/db03/oracle/rdbms9i/data/mmarttemp.ora' SIZE 50M;

   Or For Locally Managed extents:

   CREATE TEMPORARY TABLESPACE MMTEMP TEMPFILE '/db03/oracle/rdbms9i/data/mmarttemp.ora' SIZE 50M

   EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1M;

   Note: For more details about syntax and options regarding tablespace creation, see the appropriate Oracle documentation.

6. Create an Oracle user with DBA privileges to be used by the CA ERwin Data Modeler Workgroup Edition Installer or designated schema owner.

   Assign the data tablespace as this user's default tablespace, and the temporary tablespace as this user's temporary tablespace.

   For example:

   CREATE USER STEVE IDENTIFIED BY STEVE

   DEFAULT TABLESPACE MyMart

   TEMPORARY TABLESPACE MMTEMP

   QUOTA UNLIMITED ON MyMart

   QUOTA UNLIMITED ON MMARTINDEX;

7. Create the CA ERwin Data Modeler Workgroup Edition Installer role.

   The following example is the role required by the Oracle user installing CA ERwin Data Modeler Workgroup Edition.

   CREATE ROLE MMINSTALL;

8. Grant Oracle privileges to the Installer role.

   The following example shows the Oracle privileges that the CA ERwin Data Modeler Workgroup Edition Installer needs to install on Oracle. For the last command, you must log in as sys with the sysdba role in the user-name or the command will fail.

   ```
   grant create sequence to MMINSTALL;

   grant create table to MMINSTALL;

   grant create view to MMINSTALL;

   grant drop public synonym to MMINSTALL;

   grant create public synonym to MMINSTALL;

   grant create procedure to MMINSTALL;

   grant select on dba_data_files to MMINSTALL;
   ```

9. Create the CA ERwin Data Modeler Workgroup Edition User role.

   For example:

   ```
   CREATE ROLE MMUSER;
   ```

   When you select this role as the CA ERwin Data Modeler Workgroup Edition User role during Step 4 of the installation procedure, the Setup program generates grant statements that grant object level privileges to this role.

10. Grant the create session Oracle privilege to the User role.

    For example:

    ```
    grant create session to MMUSER;
    ```

    Note: The create session privilege is the only privilege that an Oracle user needs to use the database.

11. Grant the CA ERwin Data Modeler Workgroup Edition User role to each Oracle user that uses the database.

    For example:

    ```
    grant MMUSER to USER1;

    grant MMUSER to USER2;

    grant MMUSER to USER3;
    ```

    Important! You must specify the User role name created in Step 9 for these Grant statements.

12. Grant the CA ERwin Data Modeler Workgroup Edition User role to the Installer role.

    For example:

    ```
    grant MMUSER to MMINSTALL;
    ```

13. Grant the CA ERwin Data Modeler Workgroup Edition Installer role to the CA ERwin Data Modeler Workgroup Edition Installer user. You must also grant DBA privileges to the user you designate as the Installer.

    For example:

    ```
    grant MMINSTALL to STEVE;

    grant DBA to STEVE;
    ```

    Note: The DBA role is necessary only during the installation or upgrade. You can strip the installer of the DBA role after the installation or upgrade is complete.

14. Run the Setup program.

    When the setup is complete, select the Initialize option and connect to Oracle as the CA ERwin Data Modeler Workgroup Edition Installer user (for example, STEVE).

15. Select the CA ERwin Data Modeler Workgroup Edition tablespaces and User role.

    When prompted for tablespace and role information, select the CA ERwin Data Modeler Workgroup Edition data tablespace, the CA ERwin Data Modeler Workgroup Edition index tablespace, and the CA ERwin Data Modeler Workgroup Edition User role (the role created in Step 9).

16. Add the Installer user as the CA ERwin Data Modeler Workgroup Edition administrator.

    When the Security Manager is opened, your user name is automatically added to the Administrator profile in the Security Profile list. You can also designate one of the users with the MMUSER role (from step 12) for routine operations. This completes the setup and you can begin using the software.

    Note: After successful installation, you can revoke the Installer role and grant the User role to the CA ERwin Data Modeler Workgroup Edition Installer user.

Database Objects Installation

Install the software on an Oracle DBMS to create the following database objects:

- Tables
- Indexes
- Stored procedures
- Public synonyms

Oracle Connections

Oracle SQL*Net is used for client access to the database server. Although Oracle provides two utilities (SQL*Net Configuration tool and Network Manager tool) for defining your network, clients, and server, you need a network administrator to perform this task. The server system administrator is usually the best candidate.

Note: For more information about these utilities, see the Oracle *SQL*Net Administrator's Guide*.

SQL*Net requires certain files on both the server and client. These files can be generated by the Oracle networking tools, but can also be created using a text editor. Because Oracle does not support those sites that generate these files manually, it is better to use the appropriate Oracle utilities. The following lists the files required by SQL*Net:

File Name	Required On	File Contents
TNSNAMES.ORA	Client and Server	A list of service names and connect descriptors for network destinations (tells the client where it can make connections)
SQLNET.ORA	Client and Server	A list of optional diagnostic parameters
LISTENER.ORA	Server only	A list of names and addresses of all listeners on a computer and the Oracle SIDs for the databases known on that computer

Oracle SQL Commands

If a graphical DBMS access tool is not available, you can use Oracle SQL commands through SQL*DBA or SQL*Plus.

Example: Create a tablespace using the CREATE TABLESPACE command

```
CREATE TABLESPACE mm_data              /* The tablespace name. */
DATAFILE 'C:\ORANT\DATABASE\mmdata.dat'/* The data file name. */
SIZE integer_value K or M or G    /* The data file size */
DEFAULT STORAGE (              /* The default storage parameters */
    INITIAL integer_value K or M or G          /* The initial extent size */
    NEXT integer_value K or M or G             /* The next extent size */
    PCTINCREASE integer_value                  /* The percent to grow extents */
    MINEXTENTS integer_value  /* The minimum number of extents */
    MAXEXTENTS integer_value/* The maximum number of extents */
);
```

Example: Create an Oracle user using the CREATE USER command

```
CREATE USER mm_user_1   /* The user id name */
IDENIFIED BY password   /* The user password */
DEAFULT TABLESPACE tablespace_name   /* The user's default tablespace */
TEMPORARY TABLESPACE tablespace_name   /* The user's temporary tablespace */
QUOTA unlimited_or_integer_K_M_G ON tablespace_name /* The user's quota on a tablespace */
;
```

Example: Grant user privileges using the GRANT command

```
GRANT role_or_privilege_name
TO user_or_role
;
```

Mart Creation Requirements

To create the mart, you must meet one of the following requirements:

- You must be the database owner (dbo) in the target database on the Microsoft SQL Server 2000/2005/2008 or Sybase server

  Note: The database owner (dbo) is necessary only during the software installation or upgrade. It is not necessary after you create the mart.

- You must be the database schema owner user and have the DBA role in the target database on the Oracle server.
- You must have the Initializer license installed on the computer on which you want to create the mart.

Server Connection Software

The software you use to connect your client workstations with your server depends on the type of DBMS you are using and your personal preference. The following lists some commonly used DBMS client software connection packages:

DBMS	Suggested Software
Microsoft SQL Server or Sybase	Microsoft SQL Server client installation, Open Client/C Developer's Kit for PC Windows and Net-Library for PC Windows, CT-LIB, or equivalent software
Oracle	Oracle “Required Files,” SQL*Net, and TCP/IP or equivalent communication software

Note: For more information about memory and disk space requirements, see your DBMS documentation.

Connection Prerequisites

You connect to the mart on the server through the client software contained in CA ERwin Data Modeler Workgroup Edition or CA ERwin Data Modeler Navigator Edition. You must perform the following tasks before you can connect to the mart through the client:

- The administrator of the relational database management system (DBMS) must give users permission to access the DBMS on which the mart database is set up.
- The mart administrator must install the administrative files and Initializer license on the server, and set up the database. The administrator also defines user security permissions and sets up the library structure in which models are organized. When you install the CA ERwin Data Modeler Workgroup Edition software, the administrator is automatically granted Administrator security status. This lets the administrator assign user permissions and perform all required maintenance or administrative tasks.
- At least one of the client programs must be installed on the client workstation. System requirements for the client workstations are the same as those for the CA ERwin Data Modeler Workgroup Edition client software. You can find the requirements in the documentation provided with these products.

  Note: CA ERwin Data Modeler Workgroup Edition users can only connect to a mart initialized with the same serial number identifier that the user has licensed on their local PC.
- The client connectivity software for your DBMS on the server must be installed on the client workstation so that you can connect to the DBMS on which the database resides.

Refresh the Mart After Upgrading Your License

After you upgrade the license, you must refresh the mart.

To refresh the mart

1. Click Mart, Connection on the File menu.

   The Connection Manager dialog opens.
2. Connect to the database where you upgraded the license.
3. Click Close.

   The database refresh is automatic.

Create the Mart

After you create the mart database on your DBMS and install the product, you must create the mart to create the tables, install the stored procedures that manage client workgroup models in the mart, and prepare the mart for use with the clients (CA ERwin Data Modeler or CA ERwin Data Modeler Navigator Edition). To create, update, or delete a mart in CA ERwin Data Modeler Workgroup Edition, you must have the following database rights:

- For Microsoft SQL Server 2000/2005/2008, you must be the database owner (dbo).

  Notes:

  - The dbo is no longer required to have the sysadmin role.
  - The SQL Server 2005/2008 TRUSTWORTHY database property is no longer required for the mart. After you install this product, you can optionally reset this property to OFF.

For Oracle, you must be the database schema owner and have the DBA role.

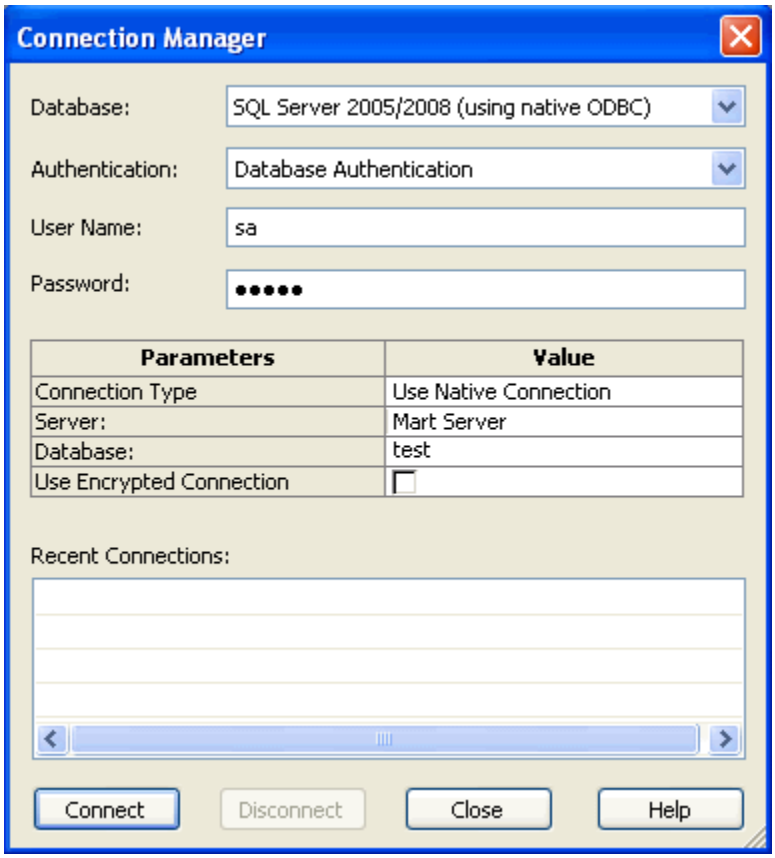
Note: The mart and the CA ERwin Data Modeler Workgroup Edition Control Tables (m7Master and m7License) must be installed on the same database for Microsoft SQL Server or Sybase DBMS.

Important! You must have the CA ERwin Data Modeler Workgroup Edition and Initializer license files present on the machine before you create the mart.

To create the mart

1. Click Mart, Connection on the File menu to log on to your DBMS that contains the mart. You must log on as the dbo or schema owner.

   The Connection Manager dialog opens.



2. Complete the following information:

Database

Identifies the type of relational database management system (DBMS) to which you are connecting. Select from the current list of supported databases.

Authentication

Windows Authentication

Specifies the use of Windows user names and passwords to secure database access.

Database Authentication

Specifies the use of a local user name and password for the connection.

Parameters/Value Options

Connection Type (Microsoft SQL Server 2005/2008 Only)

Specifies the use of Native Connection to connect using the SQL Server Native client software or ODBC data to connect using the ODBC data source you have defined.

Server

Identifies the server name.

Database

Identifies the name of the database or mart.

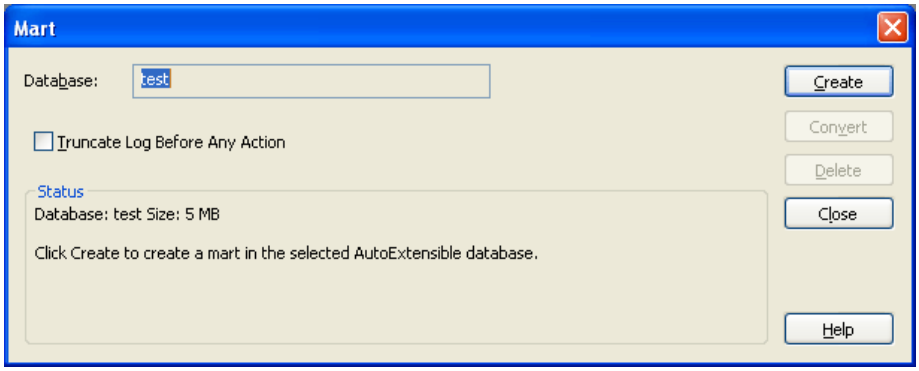
Connection String (Oracle Only)

Specifies the connection string (TNSNames entry).

You can select a database connection from the Recent Connections panel to populate the Database or Connection String previously used automatically.

3. Click Connect.

The Mart dialog opens.



4. Select from the following options:

   Database (SQL and Sybase DBMS)

   Identifies the master database name to create the tables and stored procedures.

   CA ERwin Data Modeler Workgroup Edition Role (Oracle DBMS)

   Identifies the Oracle MMUSER security role. This role is created in the procedure for creating the mart database on Oracle. Other user roles can connect to the mart, but they will encounter access problems.

   Table Tablespace

   Indicates where you want to store the mart tables.

   Note: The drop-down box is not active if there is only one tablespace.

   Index Tablespace

   Indicates where you want to store the mart indexes.

   Note: You can store tables and indexes in the same tablespace or in different tablespaces.

   Truncate Log Before Any Action

   Specifies to remove the database transaction log. Truncating the database transaction log is especially important for Microsoft SQL Server and Sybase users since a large transaction log can cause the database to hang.

   Note: When you initialize a Microsoft SQL Server 2008 database, it is not necessary to select the "Truncate Log Before Any Action" option. If you select this check box, and click Create, the following error message displays: "Incorrect syntax near the keyword 'TRANSACTION'." To avoid this message, clear the check box for "Truncate Log Before Any Action" before you click Create.

5. Click Create.

   The objects are created and the Security Manager dialog opens.

   Important! If you do not see the Security Manager dialog, it can be hidden behind other open applications. Minimize all applications or use the Alt-Tab keys until you find it.

   Your user name is automatically added to the Administrator profile in the Security Profile list.

6. Use the controls in the Security Manager dialog to assign additional security profiles. Click OK when complete.

   Note: You can also assign security profiles after you create the mart.

   The Setup Complete message opens.

7. Click OK.

   The mart is created.

Chapter 3: Upgrade

This section contains the following topics:

Software Upgrades

New versions of the software and supported client software products are periodically released. The upgrade process prepares the tables and stored procedures to accommodate new functionality and client functionality, but does not change or delete any data stored in the mart. To upgrade the software, you must be assigned to the Administrator security profile and logged on to the DBMS.

To create, update, or delete a mart in CA ERwin Data Modeler Workgroup Edition, you must have the following database rights:

- For Microsoft SQL Server 2000/2005/2008, you must be the database owner (dbo).

  Notes:

  - The dbo is no longer required to have the sysadmin role.
  - The SQL Server 2005/2008 TRUSTWORTHY database property is no longer required for the mart. After you install this product, you can optionally reset this property to OFF.
- For Oracle, you must be the database schema owner and have the DBA role.

The software should be installed or upgraded on the computer that houses the license information. This workstation is typically the same computer that the mart administrator uses or the first computer that was used to install the software.

Note: For some software upgrades, a new license is required.

The mart upgrade process is slightly different depending on your DBMS. For more information about specific procedures, see the particular section on the database you are running.

Conversion Process

The conversion systematically converts a mart, including all CA ERwin Data Modeler Workgroup Edition properties and all client models.

The time it takes to convert a mart varies for each installation. Server speed, size, complexity, and the number of models in the mart are several factors that influence conversion time.

Property Conversion

The conversion process upgrades the following CA ERwin Data Modeler Workgroup Edition properties:

- All user names and security profiles (even if the user no longer exists in the database)
- All session information
- All libraries including the preservation of the complete library structure

Access information is not converted. Access information is stored internally for tracking purposes.

Model Property Conversion

The model conversion process converts all CA ERwin Data Modeler model properties including the following properties:

Model data

CA ERwin Data Modeler models in the mart are converted. The converted models are the same as if opening an .erwin file with the upgraded client and saving the model to the mart.

All model objects

CA ERwin Data Modeler accepts all model objects from earlier versions.

All object properties

All model properties such as object names and user-defined properties (UDPs) are fully retained.

Conversion Methods

The following conversion methods are available to convert your existing client models:

Manual Conversion

Moves your models from the old mart to the new mart, one model at a time.

Automated Conversion

Installs the new software version and initializes the mart in the same location as the old mart. Click Convert to automatically convert all existing models in the old mart to the new mart.

Manual Conversion

Manual conversion means moving your models from your old mart to your new mart, one model at a time. Some advantages to using the manual method include:

- Selectively choosing the models you want to move, and leaving those that you do not.
- Optimizing the mart if there are many models or versions that are no longer in use, and that would extend conversion time.
- Running both CA ERwin Data Modeler Workgroup Edition versions simultaneously, and moving incrementally, model by model, to the new mart.
- Hosting your new mart on a different server or database platform than your old mart version.

The disadvantage is that because it is a manual process, it can be time-consuming and labor-intensive.

Automated Conversion

The automated conversion uses the Update option on the Mart dialog. Your existing CA ERwin Data Modeler models in the old mart are converted to the new mart. All models are converted automatically at the same time.

The main advantage to this method is that it is automated. The disadvantage is that it takes a long time if your mart is large, or your client/server hardware is limited. The automated conversion must be done in the same database instance as your previous version. This may require more space than is available, or require you to move your old version of the mart, if you cannot create additional space.

How to Upgrade the Mart

It is important to run through the upgrade process in a test environment before attempting to upgrade your production environment. By doing so, you become aware of any challenges that exist, and have fewer surprises when you actually convert your production environment. To convert your mart as efficiently and as fast as possible, follow these recommendations:

- Make a full backup of the current database using the backup utility for your DBMS before upgrading.
- Open and save all models in your mart using the current version of the client before you convert to CA ERwin Data Modeler Workgroup Edition r8.
- Ensure that all users check in any off-line models and ensure that there are no active sessions on the mart and that no models are checked out.
- Remove obsolete models in the current database to reduce the size, which improves the conversion speed.
- Verify that CA ERwin Data Modeler models are synchronized if you synchronize data.
- Verify that your client version is compatible with the database version to which you are migrating.
- Increase the size of the database to a minimum of three times the size of the previous database for temporary conversion objects.
- Verify there are no outstanding technical support issues related to CA ERwin Data Modeler Workgroup Edition.
- Turn on AUTOEXTEND to automatically extend the CA ERwin Data Modeler Workgroup Edition data file when full (Oracle DBMSs only).
- Make sure the Auto grow file is turned on, with your choice of File Growth and unrestricted File Growth for both data files and transaction logs (SQL Server 2000 DBMSs only).
- Review the Release Notes file.
- Run the upgrade during off-peak hours.
- Truncate the database log file.
- Start the conversion from the server to eliminate the possibility of network speed becoming a bottleneck. Have at least 1 GB on the client, typically more memory means faster performance.
- Log on to your DBMS as either the dbo (Microsoft SQL or Sybase) or schema owner (Oracle) of the database that contains the mart.

Conversion to CA ERwin Data Modeler Workgroup Edition r8

Review the following information if you are upgrading to r8:

- It is not required that you uninstall an earlier version of the product (before r8, such as CA ERwin Model Manager). Because r8 is installed to a new folder, you can continue to work with the earlier version, if you have sufficient disk space on your PC for both versions. However, when you upgrade to r8, you must be at r7.x or later in order to convert existing models to r8. Additionally, before you can use the Submodel Manager to work with models in the mart, you must first upgrade each r7.x model to r8. It is also recommended that if you plan to uninstall the r7.x program, that you create backups of your models before upgrading to r8.
- When you do upgrade to r8, verify that the mart is r7.x or later. If it is a release earlier than r7.x, you must upgrade to r7.x before you can upgrade to r8. We recommend you upgrade to the latest service pack for CA ERwin MM r7.3, and then open and save your models older than r7.x in CA ERwin MM r7.3.x before you upgrade to CA ERwin Data Modeler Workgroup Edition r8. Otherwise, the r8 upgrade process for CA ERwin Data Modeler Workgroup Edition detects whether you have Version 4.x models stored in an r7.x mart. A dialog opens that contains a list of all the Version 4.x models. You can select one of the following options for each Version 4.x model:
  - Delete the Version 4.x model and continue with the mart upgrade.
  - Cancel the mart upgrade. Doing so lets you go back to CA ERwin Data Modeler r7.x, convert the models from Version 4.x to r7.x, and save to the mart. After you have saved these models to your mart, you can continue with the r7.x to r8 upgrade.

Microsoft SQL Server or Sybase DBMS Upgrade Considerations

On a Microsoft SQL Server or Sybase DBMS, if you are updating from a previous version and the system logs are full, you may not be able to upgrade your previous installation with the current version. Before you install the new version, use a DBMS utility (such as Microsoft SQL Object Manager) and type the appropriate command to purge the transaction log.

Oracle DBMS Upgrade Considerations

On an Oracle DBMS, before you start the conversion, it is important that you turn on AUTOEXTEND to automatically extend the CA ERwin Data Modeler Workgroup Edition data file when full.

Install the Software Upgrade

To upgrade the software to a new version you must log in to your DBMS. To create, update, or delete a mart in CA ERwin Data Modeler Workgroup Edition, you must have the following database rights:

- For Microsoft SQL Server 2000/2005/2008, you must be the database owner (dbo).

  Notes:

  - The dbo is no longer required to have the sysadmin role.
  - The SQL Server 2005/2008 TRUSTWORTHY database property is no longer required for the mart. After you install this product, you can optionally reset this property to OFF.

- For Oracle, you must be the database schema owner and have the DBA role.

To install the software upgrade

1. Close all client connections and either insert the installation DVD or double-click the file you downloaded from the online CA product page.

   The Installation Wizard opens.

2. Follow the prompts in the wizard to proceed to perform an upgrade.

   When the install completes, a final screen displays.

3. Click Close.

Note: You may be required to reinitialize the mart to complete the upgrade process. For more information see Convert or Update a Mart (see page 41).

Convert or Update a Mart

The model is not fully converted or updated until you initialize the mart and initially open and save the model in the mart with CA ERwin Data Modeler.

Note: When you first open the model, it can take a few minutes longer to complete the process.

To convert a mart

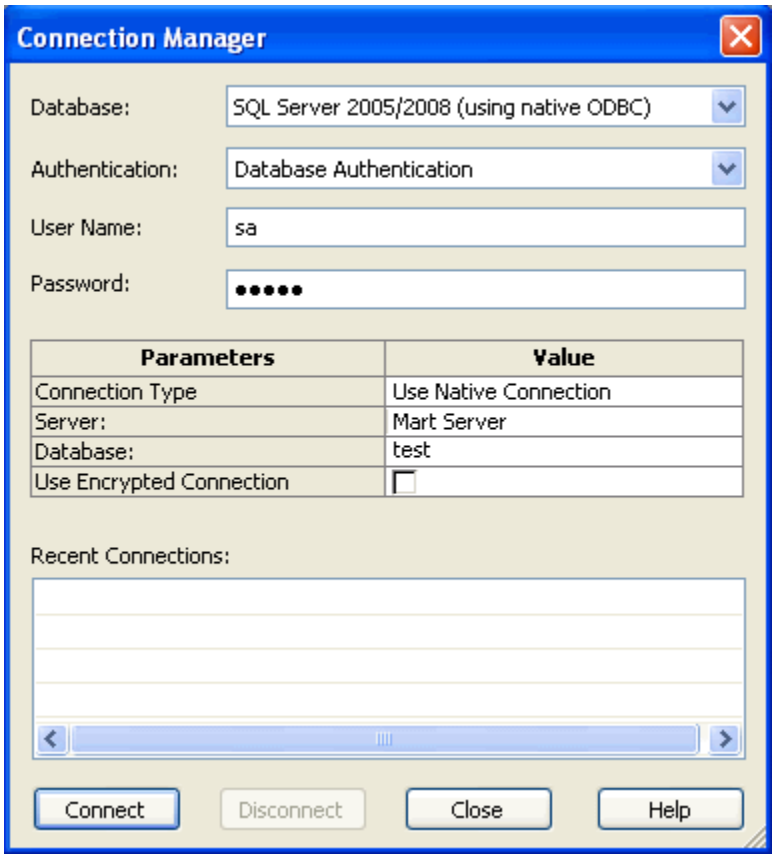
1. Log on to your DBMS that contains the mart. To create, update, or delete a mart you must have the following database rights:
   - For Microsoft SQL Server 2000/2005/2008, you must be the database owner (dbo).

     Notes:

     - The dbo is no longer required to also have the sysadmin role.
     - The SQL Server 2005/2008 TRUSTWORTHY database property is no longer required for the mart. After you install this product, you can optionally reset this property to OFF.
   - For Oracle, you must be the database schema owner and have the DBA role.

2. Click Mart, Initialize Mart on the File menu.

   The Connection Manager dialog opens.



3. Complete the following information:

   Database

   Identifies the type of relational database management system (DBMS) to which you are connecting. Select from the current list of supported databases.

   Authentication

   Windows Authentication

   Specifies the use of Windows user names and passwords to secure database access.

   Database Authentication

   Specifies the use of a local user name and password for the connection.

Parameters/Value Options

Connection Type (Microsoft SQL Server 2005/2008 Only)

Specifies the use of Native Connection to connect using the SQL Server Native client software or ODBC data to connect using the ODBC data source you have defined.

Server

Identifies the server name.

Database

Identifies the name of the database or mart.

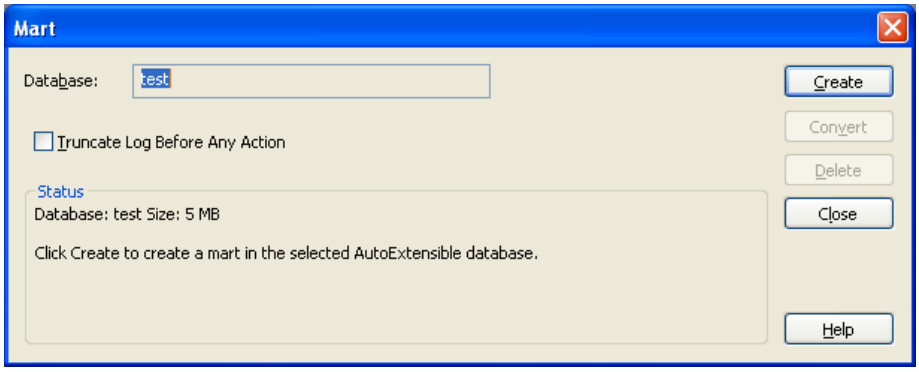
Connection String (Oracle Only)

Specifies the connection string (TNSNames entry).

You can select a database connection from the Recent Connections panel to populate the Database or Connection String previously used automatically.

4. Click Connect.

The Mart dialog opens.



5. Select from the following options:

   Database (SQL and Sybase DBMS)

   Identifies the master database name to create the tables and stored procedures.

   CA ERwin Data Modeler Workgroup Edition Role (Oracle DBMS)

   Identifies the Oracle MMUSER security role. This role is created in the procedure for creating the mart database on Oracle. Other user roles can connect to the mart, but they will encounter access problems.

   Table Tablespace

   Indicates where you want to store the mart tables.

   Note: The drop-down box is not active if there is only one tablespace.

   Index Tablespace

   Indicates where you want to store the mart indexes.

   Note: You can store tables and indexes in the same tablespace or in different tablespaces.

   Truncate Log Before Any Action

   Specifies to remove the database transaction log. Truncating the database transaction log is especially important for Microsoft SQL Server and Sybase users since a large transaction log can cause the database to hang.

   Note: When you initialize a Microsoft SQL Server 2008 database, it is not necessary to select the "Truncate Log Before Any Action" option. If you select this check box, and click Create, the following error message displays: "Incorrect syntax near the keyword 'TRANSACTION'." To avoid this message, clear the check box for "Truncate Log Before Any Action" before you click Create.

6. Select one of the following after verifying the mart name in the Status window:

   Create

   Creates an empty mart. You can manually open and save existing CA ERwin Data Modeler models to this new mart.

   Update

   Creates a new mart with the current version and updates the data from the old mart to the new mart. The Update button is enabled only when an earlier version of the software is detected. When you update your mart, the old mart is no longer usable in the previous version.

   Note: The Create and Update buttons are mutually exclusive and the text of the button changes depending if a previous version of the mart is detected. If the Create button is not enabled, then a previous version of a mart is detected.

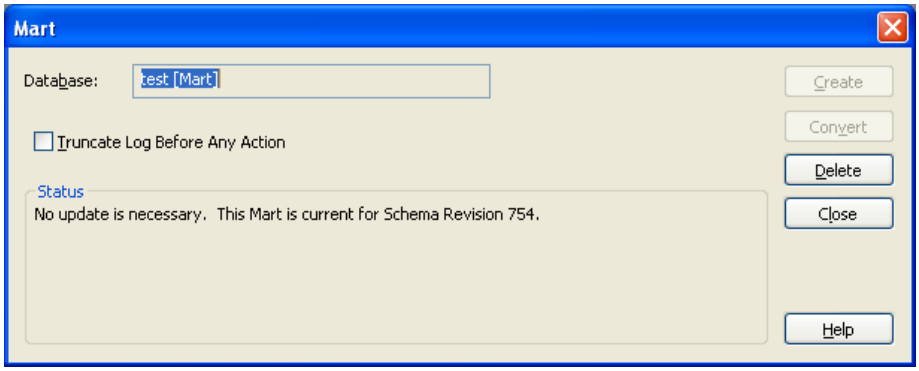
   Convert

Creates a new mart with the current version and copies the data from the old mart to the new mart. The old mart is retained in the same database.

The dialog displays conversion progress messages.

7. Click OK in the Upgrade has Successfully Completed message dialog.

   The Mart dialog displays the status.



8. Click Close.

   The administrative setup is complete.

9. Click OK on the Setup Complete dialog.

   The conversion is complete.

The same user profile assignments exist as before the conversion. If you must make changes to user profiles, click Mart, Security Manager on the File menu to open Security Manager.

Appendix A: Microsoft SQL Server and Sybase Tuning Recommendations

This section contains the following topics:

Physical Tuning Parameters

The mart is dynamic, with many queries and data manipulations performed with each model save and load. Although the Microsoft SQL Server and Sybase architectures differ in some ways, the basic operation and configuration of both server environments is the same.

The configuration changes that must be made to Microsoft SQL Server and Sybase for optimum performance with maximum fault tolerance and recoverability are provided.

Hardware Configurations

Consider implementing the mart for CA ERwin Data Modeler Workgroup Edition on a multiprocessor system, especially if multiple data modelers are saving information simultaneously. Install the software on a stand-alone Microsoft SQL Server, if possible, or on a development computer. The database overhead noticeably affects other databases running on the same system during save and load operations. If few modelers save and load simultaneously, the overhead is less.

To improve caching performance, implement the mart on servers that have at least 512 MB of RAM.

Database Placement

CA ERwin Data Modeler Workgroup Edition mart databases are written to in bursts of activity. Stored procedures and bulk insert statements are used to minimize overhead when saving data to Microsoft SQL Server. To optimize performance, place the database on low activity Microsoft SQL Server files. If possible, place the log and data segments on different physical drives to reduce contention between log writes and database reads.

Data Redundancy

Make sure that you make a copy of the transaction log on a separate drive to maximize recoverability of the database. If the up-time of the database is considered critical, make a copy of the database and the transaction logs, and other Microsoft SQL Server files you consider important.

Server-Level Tuning Parameters

You should perform the following server-level tuning tasks:

- Configure Microsoft SQL Server to use as much memory as possible. Allocating more memory to the Microsoft SQL Server caching mechanisms means less physical reads from the disk and improved database query performance.
- Allow several megabytes of disk space for the procedure cache because CA ERwin Data Modeler Workgroup Edition uses many stored procedures.

The following table shows the recommended configuration parameters for a Microsoft SQL Server running CA ERwin Data Modeler Workgroup Edition:

Parameter Type	Recommended Value	Notes
memory (Sybase 11: "total memory")	512 MB minimum **Note:** 1 GB is recommended	More memory implies less physical I/O
procedural cache	Set the cache to grow to a fixed size, such as 50 MB or 100 MB.	CA ERwin Data Modeler Workgroup Edition is stored procedure-intensive

Named Cache Configuration

For Sybase systems, consider setting up a 4 KB pool for the default data cache. Sybase writes I/O to the log more often in 4 KB increments than in 2 KB increments (the default). You can set up a 4 KB I/O pool using sp_poolconfig, but you must restart the server to enable the 4 KB I/O writes to the log. Set the pool up as a smaller subset of the cache.

Note: Be careful when making changes to the data caching systems in Sybase. Monitor your changes with SQL Monitor or sp_sysmon to ensure that the changes you make do not starve the 2 KB I/O pool or other caches.

The following shows the recommended size of the 4 KB I/O pool for small, medium, and large servers:

Server Size	Data Cache Size	4 KB I/O Pool Size
Small	128 MB	4 MB
Medium	512 MB	6 MB
Large	1 GB	10 MB

Database Tuning Parameters

You can make database tuning adjustments in three different areas:

- Transaction log
- Threshold procedures
- Database options

Transaction Log

The transaction log keeps a before and after image of each change made in the database. Microsoft SQL Server keeps a transaction log for each database.

Back up the transaction log frequently. This keeps the transaction log small and reduces the amount of data lost in the event of a severe database corruption.

Threshold Procedures

For Sybase, enable a threshold procedure for the last-chance threshold to back up the log when it runs out of disk space.

Note: Because Microsoft SQL Server does not provide such an option, you must back up the log frequently to keep the log small.

The following shows a sample Sybase threshold action procedure:

```
create procedure sp_thresholdaction
@dbname char(40),
  @segment_name char(40),
  @space_left int,
  @status int
as
/* make the thresholdaction procedure backup the log */
declare @backdevice varchar(255)
  select @backdevice= '/u/backups/tranfile'+
          convert(char(8),getdate(),4)
  dump transaction mart to @backdevice
go
```

"trunc. log on chkpt" Option

To ensure maximum recoverability, do not enable the “trunc. log on chkpt” option, since the log is cleared automatically after each checkpoint operation. If the database device becomes damaged with the log device still active and this option is set, the database cannot be recovered using the log files since they are almost empty.

Database and Log Sizing

Since the database is a dynamic environment, allow plenty of space for the models you create. Tests have shown that models generate between 10 KB and 20 KB of data per entity during initial save times. Log overhead per object is approximately 15 KB to 21 KB per entity. Allocate 50 to 65 percent of the database size to the log to avoid running out of room in the log for a typical model save.

An average mart can range in size between 200 MB and 1 Gig. Size the transaction log accordingly. A 50 MB data device can store several large models (about 2,500 entities and 100,000 total objects), but keep in mind that the database becomes more flexible and has less storage space issues when the data device is large.

The following table shows the recommended database size and log size for small, medium, and large marts:

Database Size	Database Device	Transaction Log Device
Small	200 MB	100 MB
Medium	500 MB	200 MB
Large	1 GB	400 MB

How to Maintain an Efficient Database

To maintain an efficient database, perform the following maintenance tasks:

- Run UPDATE STATISTICS and execute sp_recompile frequently on every table in the database. This keeps the statistics up-to-date for the indexes, resulting in better overall performance.
- Periodically recreate the clustered indexes in the database to reduce fragmentation. Be sure to back up the database and transaction log daily.
- Run DBCC CheckDB(), DBCC CheckCatalog(), and DBCC CheckAlloc() or DBCC NewAlloc() on the database nightly, to check for corruption and inconsistencies in the database. Check the output of these queries and look for keywords like corrupt. Any problems detected by these commands are sent to the query output, so save the files and scan them regularly.

The following table shows maintenance tasks and the recommended frequency for performing these tasks:

Maintenance Task	Recommended Frequency	Reason
DBCC Checkdb	Nightly	Check for corruption in databases.
DBCC NewAlloc	Weekly	Check for allocation corruption (Microsoft SQL Server only).
DBCC CheckAlloc	Weekly	Check for allocation corruption (Microsoft SQL Server only).
DBCC CheckCatalog	Weekly	Check for system table inconsistencies.
UPDATE STATISTICS	Nightly	Recreates the statistics page for each index.
EXEC sp_recompile	Nightly	Tells which stored procedures have changed.
Backup Database	Weekly	Full backup of database should be done at least weekly, if not nightly.
Backup Transaction Log	Daily	Backing up the transaction log daily saves all committed transactions and clears the log.

Nightly Maintenance Script

The following script is the recommended nightly maintenance script:

```
UPDATE STATISTICS m7Object
go
UPDATE STATISTICS m7ObjectProperty
go
```

DBCC Commands

Use the following DBCC commands to perform maintenance tasks on the database:

```
DBCC CheckDB(mart)
 go
DBCC CheckAlloc(mart)
 go
DBCC CheckCatalog(mart)
 go
```

Appendix B: Oracle Tuning Recommendations

This section contains the following topics:

Recommended Disk Configurations

Oracle servers are available on many different platforms ranging from PCs to mainframes. To support the wide range of hardware platforms and application requirements, Oracle servers have a number of configuration options that you can use to tailor the behavior of an Oracle database. You can also use these configuration options to tailor your database. Some of the options documented in this guide only apply to specific Oracle versions. For further information, see the appropriate Oracle documentation.

If the Oracle server is not dedicated to the CA ERwin Data Modeler Workgroup Edition mart, as database administrator you must be careful to balance the requirements of all users on the server. This is particularly true for database servers that support OLTP (On Line Transaction Processing) applications that require tight response characteristics.

The layout of Oracle database files on physical disk drives plays a major role in the performance of the database. When used with CA ERwin Data Modeler Workgroup Edition, Oracle performs large bursts of database operations (fetches, inserts, updates, and deletes) in a short period of time. This usage pattern often requires Oracle to perform a significant amount of disk I/O operations. For good performance, it is important that as many I/O operations as possible be performed in parallel. This enables Oracle to retrieve data faster during fetches and to write data faster during inserts, updates, and deletes.

To maximize parallel I/O access, split the Oracle database files across many physical disks. Ideally, you should place the database on four physical disks. If this is not possible, you can use three, two, or one disk systems.

Note: Installing Oracle on a single disk system causes significant performance degradation.

Database Configuration

Proper database configuration is essential to the smooth operation and good performance of an Oracle database. You can use the configuration options described to optimize the processing of your Oracle server.

Maximum Number of Processes

Oracle must be preconfigured at startup with a maximum number of processes that can access the database. The parameter that controls the maximum number of processes, called PROCESSES, is defined in the INIT.ora or INIT<SID>.ora file. It should be set to the maximum potential number of concurrent users plus seven. The additional seven are for Oracle background processes that must also access the database. Setting the number of processes at a lower value can prevent some users from accessing the server during peak usage times. Setting the number of processes at a higher value can waste a small amount of system memory resources.

Shared Pool

The shared pool is an area of Oracle memory that includes two main structures:

- The library cache, which stores parsed SQL and PL/SQL statements
- The dictionary cache, which stores the Oracle data dictionary (or Oracle metadata)

The SHARED_POOL_SIZE parameter is defined in the INIT.ora or INIT<SID>.ora file and is used to regulate the size of the shared pool. Set the SHARED_POOL_SIZE parameter to a minimum of 3500000 (3.5 million). Setting the shared pool to a smaller value can degrade performance, forcing Oracle to do disk I/O to retrieve objects that cannot fit in the shared pool. A larger shared pool may be required, depending on the size and number of models and the number of users.

Buffer Cache

The buffer cache serves as a memory cache for all data going to and from Oracle data files. When Oracle needs a block of data it first checks whether that block exists in the buffer cache. If it does, Oracle gets the data from the buffer cache—avoiding disk access. Having a large enough buffer cache lets the Oracle server bypass most I/O requests.

Buffer cache size is controlled by the DB_BLOCK_BUFFERS parameter defined in the INIT.ora or INIT<SID>.ora file. Set the buffer cache to a minimum of 4 MB. Values smaller than 4 MB force Oracle to do many more I/O requests and significantly degrades the performance. Values larger than 4 MB improve performance and should be used if memory is available.

Note: In Oracle, the DB_BLOCK_BUFFERS parameter is specified as a number of database blocks instead of actual size in bytes. To compute the value of DB_BLOCK_BUFFERS, divide the desired buffer cache size in bytes by the database block size defined by the DB_BLOCK_SIZE parameter in the INIT.ora or INIT<SID>.ora file.

Redo Log Files

Redo log files contain a record of all Data Manipulation Language (DML) commands (such as INSERT, UPDATE, and DELETE commands) performed on the database. As DMLs are performed, the Oracle engine writes them to sequential redo log files. Periodically, an Oracle background process retrieves the DMLs from the Redo log files and writes the actual changes to the Oracle tablespace files. This mechanism lets Oracle defer most of the I/O burden associated with DMLs to a background process that does not slow down the client processes.

Note: Oracle Redo log files are treated by the database engine as a ring. When one file fills up, the engine performs a log switch and starts writing to the next log file in the ring. When that log file fills, the engine switches again.

The number and size of the Redo log files is an important performance consideration. If the log files are too small or if there are not enough of them in the ring, Oracle may have to stall on a log switch. If this happens, the DMLs in the next log in the ring may not have been written to the tablespace files, and therefore the next log file has not been archived by the background ARCH process. For more information about Redo log archiving, see the Oracle documentation.

Redo log files are created when the database is created. However, Redo log files can be added or deleted at any time using Data Definition Language (DDL) statements.

You should have at least four Redo log files, each 2 MB in size. A smaller number of Redo log files or a smaller Redo log file size can cause I/O bottlenecks. If the disk space is available, using more than four Redo log files further reduces the chances of delayed log switches. Larger Redo log files improve performance, but care must be taken to adequately schedule checkpoints that write DMLs stored in the Redo logs to the tablespace files.

If the number of disks permits, you should mirror Redo log files by creating Redo log groups with two mirrored members per group. This offers the database some protection against single disk errors.

Increase Space in Redo Logs

The number and size of the Redo log files is an important performance consideration. Redo log files are created when the database is created.

To increase space in the Redo logs

1. Log in to Oracle using SQL*Plus as SYSDBA or SYSOPER and run the following script to create eight Redo logs that are 2 MB each:

```
rem -- parm1 -- temp dir

rem Generate creation script
SELECT 'ALTER DATABASE ADD LOGFILE ''
     ||   SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\' , -1 , 1 ) )
     || 'REDO_11.LOG'' SIZE 2M ;'
  from V$LOGFILE
 where ROWNUM = 1
   and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_11.LOG' )
 UNION

SELECT 'ALTER DATABASE ADD LOGFILE ''
     ||   SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\' , -1 , 1 ) )
     || 'REDO_12.LOG'' SIZE 2M ;'
  from V$LOGFILE
 where ROWNUM = 1
   and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_12.LOG' )
 UNION

SELECT 'ALTER DATABASE ADD LOGFILE ''
     ||   SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\' , -1 , 1 ) )
     || 'REDO_13.LOG'' SIZE 2M ;'
  from V$LOGFILE
 where ROWNUM = 1
   and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_13.LOG' )
 UNION
```

```
SELECT 'ALTER DATABASE ADD LOGFILE ''
     ||   SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\' , -1 , 1 ) )
     || 'REDO_14.LOG'' SIZE 2M ;'
  from V$LOGFILE
 where ROWNUM = 1
   and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_14.LOG' )
 UNION

SELECT 'ALTER DATABASE ADD LOGFILE ''
     ||   SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\' , -1 , 1 ) )
     || 'REDO_15.LOG'' SIZE 2M ;'
  from V$LOGFILE
 where ROWNUM = 1
   and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_15.LOG' )
 UNION

SELECT 'ALTER DATABASE ADD LOGFILE ''
     ||   SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\' , -1 , 1 ) )
     || 'REDO_16.LOG'' SIZE 2M ;'
  from V$LOGFILE
 where ROWNUM = 1
   and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_16.LOG' )
 UNION

SELECT 'ALTER DATABASE ADD LOGFILE ''
     ||   SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\' , -1 , 1 ) )
     || 'REDO_17.LOG'' SIZE 2M ;'
  from V$LOGFILE
 where ROWNUM = 1
   and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_17.LOG' )
 UNION

SELECT 'ALTER DATABASE ADD LOGFILE ''
     ||   SUBSTR ( MEMBER , 1 , INSTR ( MEMBER , '\' , -1 , 1 ) )
     || 'REDO_18.LOG'' SIZE 2M ;'
  from V$LOGFILE
 where ROWNUM = 1
   and not exists ( SELECT 1 from V$LOGFILE where MEMBER like '%REDO_18.LOG' )
.
 Spool &1.AddLog.ORA
/
Spool Off
COMMIT ;

SELECT 'ALTER SYSTEM SWITCH LOGFILE ;' from V$LOG where ROWNUM < 5
.
 Spool &1.SwtchLog.ORA
/
Spool Off
COMMIT ;
```

```
rem Generate Deletion script
SELECT 'ALTER DATABASE DROP LOGFILE GROUP ' || TO_CHAR ( GROUP# ) || ' ;' from V$LOG
where BYTES < 2097152
.
 Spool &1.DropLog.ORA
/
Spool Off
COMMIT ;

@&1.AddLog.ORA
$Del &1.AddLog.ORA
COMMIT ;

@&1.SwtchLog.ORA
$Del &1.SwtchLog.ORA
COMMIT ;

@&1.DropLog.ORA
$Del &1.DropLog.ORA
COMMIT ;
```

2. Run the following query to verify the new Redo log configuration:

```
SQLWKS> select group#, status, bytes from v$log;
```

Note: Oracle does not let you drop an older Redo log file that is still ACTIVE. If you must drop an older Redo log file that is still ACTIVE, drop that log file manually.

Redo Log Buffer

The Redo log buffer is an area in memory that Oracle uses to collect DMLs before they are written to the Redo log files. The log file write occurs when either a transaction commits or a Redo log buffer is full. Because CA ERwin Data Modeler Workgroup Edition tends to generate fairly large transactions, use a log buffer size of 163840 bytes. (For single disk Oracle installations, use a log buffer size of 655360 bytes.) Setting the Redo log buffer size to a value smaller than the recommended value can degrade I/O performance. Using a larger value requires more memory.

The Redo log buffer size is defined using the LOG_BUFFER parameter in the INIT.ora or INIT<SID>.ora file.

Checkpoints

An Oracle checkpoint is an event that posts DMLs from the Redo log files to the tablespace files. Checkpoints always occur after a Redo log file switch and can also be configured to occur at predefined time intervals. For CA ERwin Data Modeler Workgroup Edition, which generates large transactions, checkpoints should occur only after log switches to minimize I/O.

To ensure a checkpoint only after a log switch, *do not* set the LOG_CHECKPOINT_INTERVAL and LOG_CHECKPOINT_TIMEOUT parameters in the INIT.ora or INIT<SID>.ora file.

Storage Parameters

Object storage parameters in Oracle determine the amount of space allocated for each object in the database. Setting these parameters correctly is critical to both operation and performance of the database. Incorrect storage allocations can cause a database object to run out of space, which prevents you from saving models to the repository. Inefficient selection of storage parameters can lead to performance problems by forcing Oracle to do time-consuming space management operations during DML statement execution.

Recommendations for storage parameter values differ based on the size of your model:

- Small data models: 1 to 50 Entities
- Medium data models: 51 to 100 Entities
- Large data models: over 100 Entities

Rollback Segments

Rollback segments contain undo information for all changes performed by noncommitted transactions. Rollback segments are a shared resource used by all active transactions in the database. When a transaction starts, Oracle binds that transaction to a particular rollback segment. As DMLs in the transaction execute, rollback segment space is used. For large transactions, rollback segments may need to allocate new extents as the transaction continues. When the transaction ends, a properly configured Oracle database releases the additional rollback extents so that they can be used for other rollback segments. To ensure that the additional rollback extents are released, configure the OPTIMAL parameter for each rollback segment.

Use a maximum of five transactions per rollback segment by setting the TRANSACTIONS_PER_ROLLBACK_SEGMENT parameter in the INIT.ora or INIT<SID>.ora file. Use the following rollback segment storage parameters for small, medium, and large models:

Model Type	Initial Extent Size	Next Extent Size	Optimal Size
Small	1 MB	1 MB	~ 2 MB x (# of transactions)
Medium	6 MB	6 MB	~ 12 MB x (# of transactions)
Large	6 MB	6 MB	~ 12 MB x (# of transactions)

Note: The "# of transactions" is the maximum number of simultaneous CA ERwin Data Modeler Workgroup Edition connections to Oracle.

In Oracle it is common to see the following error statement when the Rollback logs are not set up optimally:

ORA-01562 failed to extend rollback segment number string

Set Optimal Configuration for the Rollback Logs

For large transactions, rollback segments may need to allocate new extents as the transaction continues. When the transaction ends, a properly configured Oracle database releases the additional rollback extents so that they can be used for other rollback segments. To ensure that the additional rollback extents are released, configure the OPTIMAL parameter for each rollback segment.

To set the optimal configuration for the rollback logs of large databases

1. Log in to Oracle as SYS or SYSTEM and run the following query and check the result to view the current configuration of the Rollback Segments:

   SQLWKS> select SEGMENT_NAME, INITIAL_EXTENT, NEXT_EXTENT, MIN_EXTENTS, MAX_EXTENTS, STATUS from dba_rollback_segs;

   The result is:

SEGMENT_NAME	INITIAL_EX	NEXT_EXTEN	MIN_EXTENT	MAX_EXTENT	STATUS
SYSTEM	51200	51200	2	121	ONLINE
RB_TEMP	102400	102400	2	121	OFFLINE
RB1	2097152	2097152	2	121	ONLINE
RB2	2097152	2097152	2	121	ONLINE
RB3	2097152	2097152	2	121	ONLINE
RB4	2097152	2097152	2	121	ONLINE
RB5	2097152	2097152	2	121	ONLINE
RB6	2097152	2097152	2	121	ONLINE
RB7	2097152	2097152	2	121	ONLINE
RB8	2097152	2097152	2	121	OFFLINE
RB9	2097152	2097152	2	121	OFFLINE
RB10	2097152	2097152	2	121	OFFLINE
RB11	2097152	2097152	2	121	OFFLINE
RB12	2097152	2097152	2	121	OFFLINE
RB13	2097152	2097152	2	121	OFFLINE
RB14	2097152	2097152	2	121	OFFLINE
RB15	2097152	2097152	2	121	OFFLINE
RB16	2097152	2097152	2	121	OFFLINE

   18 rows selected.

   Note: In this example, there are sixteen rollback segments with 2 MB INITIAL EXTENT, 2 MB NEXT EXTENT, and MAX EXTENTS of 121 MB.

2. Run the following query for each Rollback Segment:

```
alter rollback segment rb1 offline;
 drop rollback segment rb1;
 create public rollback segment rb1
tablespace rollback_data
storage (initial 6M
next 6M
minextents 2
maxextents 121
optimal 12M);
 alter rollback segment rb1 online;
 alter rollback segment rb2 offline;
 drop rollback segment rb2;
 create public rollback segment rb2
tablespace rollback_data
storage (initial 6M
next 6M
minextents 2
maxextents 121
optimal 12M);
 alter rollback segment rb2 online;

<Repeat for each Rollback Segment>
```

This query sets the optimal Rollback Segment configuration to 6 MB INITIAL EXTENT, 6 MB NEXT EXTENT, 2 MB MIN EXTENT, 121 MB MAX EXTENT, and 12 MB OPTIMAL.

3. Run the following query to verify the Rollback segment configuration changes:

```
SQLWKS> select SEGMENT_NAME, INITIAL_EXTENT, NEXT_EXTENT,
MIN_EXTENTS,MAX_EXTENTS, STATUS from dba_rollback_segs;
```

The result is:

SEGMENT_NAME	INITIAL_EX	NEXT_EXTEN	MIN_EXTENT	MAX_EXTENT	STATUS
SYSTEM	51200	51200	2	121	ONLINE
RB_TEMP	102400	102400	2	121	OFFLINE
RB1	6291456	6291456	2	121	ONLINE
RB2	6291456	6291456	2	121	ONLINE
RB3	6291456	6291456	2	121	ONLINE
RB4	6291456	6291456	2	121	ONLINE
RB5	6291456	6291456	2	121	ONLINE
RB6	6291456	6291456	2	121	ONLINE
RB7	6291456	6291456	2	121	ONLINE
RB8	6291456	6291456	2	121	ONLINE
RB9	6291456	6291456	2	121	ONLINE
RB10	6291456	6291456	2	121	ONLINE
RB11	6291456	6291456	2	121	ONLINE
RB12	6291456	6291456	2	121	ONLINE
RB13	6291456	6291456	2	121	ONLINE
RB14	6291456	6291456	2	121	ONLINE
RB15	6291456	6291456	2	121	ONLINE
RB16	6291456	6291456	2	121	ONLINE

18 rows selected.

Tablespaces

To improve performance you can create more than one tablespace for indexes. Then, after you install the software, you can move some indexes to the other tablespaces.

The following table lists the suggested initial sizes for DATA and INDEX tablespaces:

Tablespace	Small	Medium	Large
DATA	200 MB	500 MB	1 GB
INDEX	200 MB	250 MB	500 MB

DATA and INDEX tablespaces with these initial sizes can accommodate at least ten models or versions.

Recommended Initial Table Sizes

The following table lists the recommended initial extent sizes in kilobytes for the most important tables for small, medium, and large models:

	Recommended Initial Extent Size (KB)		
Table Name	**Small**	**Medium**	**Large**
m7Object	1096	2193	10963
m7ObjectProperty	3288	6579	32889

These initial extent sizes are chosen so they can accommodate two versions of a model. These values are the recommended values of the INITIAL parameter in the STORAGE clause of the CREATE TABLE statement. The size of the next extent (NEXT parameter in the STORAGE clause) should be half of the corresponding initial size.

Recommended Initial Indexes

The following lists the recommended initial extent sizes in kilobytes for the most important indexes for small, medium, and large models:

	Recommended Initial Extent Size (KB)		
Index Name	**Small**	**Medium**	**Large**
XPKm7Object	1415	2829	14146
XPKm7ObjectProperty	2098	4195	20975

Appendix C: CA ERwin Data Modeler Workgroup Edition Performance

This section contains the following topics:

Optimize Your Performance

The performance of CA ERwin Data Modeler Workgroup Edition depends on many factors. This appendix outlines some steps you can take to improve your performance. The five main components where a slow down can occur are the database, the server, the network, the client PC, and the CA ERwin Data Modeler Workgroup Edition software.

Database
- Reindexing
- Statistics
- Tuning

Server
- RAM
- Speed (GHz)
- CPU
- Disk space
- Dedicated to CA ERwin WE or shared by multiple applications?

Network
- LAN or WAN?
- Speed
- Throughput
- Traffic

Client Machine
- RAM
- Speed (GHz)
- CPU
- Disk space and virtual memory

Software

You should regularly reindex the database and run the database statistics. Sometimes running the database statistics alone does not affect the performance, however it is good practice to run the database statistics on a regular basis to ensure the maximum performance of your database.

In past releases, multiple versions of the same model in the mart can slow down overall performance. You can remove versions by selectively purging them using the Version Manager. CA ERwin Data Modeler Workgroup Edition r7.2 or later lets you to disable the versioning so it does not create a new version with every save of a model to the mart.

If you have many domains in a model, they consume a lot of memory and file space. Having too many domains can slow down CA ERwin Data Modeler Workgroup Edition. If you need to use many domains, we suggest you create a template model to house them, and then import the specific domains that you need, to each of your individual models. If you are using the domains to enforce unique attribute or column naming standards, you can alternatively use the ERwin Glossary (*.nsm) for that purpose.

Check the CA ERwin Data Modeler Workgroup Edition Release Notes for the latest system requirements. Greater CPU speed, and more RAM and disk space result in better performance.

Reindex the Database

Over a period of time, database indexes become fragmented. A large number of inserts and deletes can lead to significant performance degradation. The scripts provided in this appendix rebuild the indexes and fix them. In some cases, it can make a significant difference in the mart performance.

We recommend that you reindex your mart database nightly or during off-peak hours. As the index is recreated, the process can temporarily degrade the performance of your database. You can create an automated batch job to do the reindex. You should also reindex the database after a large model merge, a save to the mart, and right after a conversion of the mart when upgrading to a later version of the software.

Run the Oracle DBMS Reindex Script

As the new index is being built, it coexists with the old index in the database. For this reason, you should plan for enough space to store both the old index and the new index. When the index is rebuilt, the new index becomes available, the old index is dropped, and the space is reclaimed by the database. If you encounter any errors while rebuilding the indexes, re-run the statements. If you require more space to rebuild those specific indexes, add more storage to your index tablespace, and then try rebuilding those specific indexes again.

To run the Oracle DBMS reindex script

1. Create the mmreindex.ora script and copy it locally.
2. Edit the script and replace 'MODELMART' with the name of the mart schema-owner and 'MMINDEX' with the name of the mart index tablespace. Save your changes.
3. Connect to SQL*PLUS as the user SYS.
4. Grant the 'ALTER ANY INDEX' privilege to the mart Schema Owner.
5. Run your Oracle query tool and execute the following at the SQL prompt:

   GRANT ALTER ANY INDEX TO <MART SCHEMA OWNER>;
6. Disconnect user SYS and Connect to your Oracle query tool as the Schema Owner.
7. Execute the following Script at the SQL Prompt:

   @c:\mmreindex.ora

Example: Oracle Reindex Script (MMReIndex.ora)

```
-------------------------------------------------------------------------------
-- Object:     MMReIndex.ora
-- Desc:        Use this Procedure to ReIndex the MM ORACLE Repository whenever a
--                 Merge/Save of big model is done to MM
--                 Limitation(s) is specific to Oracle Releases >= 817
-- For ORACLE DBMS < 8i Modify the script to Use NOPARALLEL
-- NOTE:    You will need to change MMOWNER to the Mart schema owner name.
--             You will need to change MMINDEX to the Mart index tablespace.
-- Oracle indexes are not self-balancing. They become fragmented after a large
-- number of INSERTs and DELETEs which may lead to significant performance degradation.
-- This script rebuilds the Mart indexes and cures them.
-------------------------------------------------------------------------------
set pagesize 1000
set linesize 2000
set verify off
set feedback off
set heading off
spool c:\mmreindex.ora
SELECT            'ALTER INDEX ' || USER || '.' || INDEX_NAME ||
' REBUILD PARALLEL NOLOGGING COMPUTE STATISTICS TABLESPACE MMINDEX;'
FROM     DBA_INDEXES
WHERE OWNER = UPPER ('MMOWNER')
AND      (INDEX_NAME like 'XPK%' or INDEX_NAME like 'XAK%'
          or INDEX_NAME like 'XIE%')
order by index_name;
spool off
set heading on
set pagesize 24
set verify on
set feedback on
@c:\mmreindex.ora
/
```

Example: Microsoft SQL Server DBMS reindex script

```
-- Drop the Procedure appropriately
IF EXISTS (SELECT name FROM sysobjects WHERE name = N'usp_ReIndex' AND type = N'P')
Begin
        DROP PROCEDURE usp_ReIndex
        Print 'Procedure Dropped'
End
GO
--------------------------------------------------------------------------------
-- Object:      usp_ReIndex
-- Desc:        Use this Procedure to ReIndex the MM SQL REpository whenever a
--              Merge/Save of big model is done to MM
-- Change History:
-- Name                 Date            Reason
--------------------------------------------------------------------------------
CREATE PROCEDURE usp_ReIndex AS
        Declare
                @Cmd    varchar(2000),
                @Name   Sysname
        DECLARE tmp_Reindex CURSOR LOCAL FOR
                SELECT Name
                FROM    SysObjects
                WHERE   Type = 'U'
       OPEN tmp_Reindex
        FETCH NEXT FROM tmp_Reindex INTO @Name
        WHILE @@FETCH_STATUS = 0
        BEGIN
                Print 'Processing Index for Table ' + @Name
                Set @cmd = 'DBCC DBREINDEX (' + @Name +   ', "", 0)'
                Exec (@Cmd)
                If @@Error <> 0
                        Print 'Error Reindexing Table ' + @Name
                FETCH NEXT FROM tmp_Reindex INTO @Name
        END
        Close tmp_Reindex
go
-- ReIndex the DB
Exec usp_ReIndex
```

Example: Sybase DBMS reindex script (MMReIndex.ora)

```
-- Drop the Procedure appropriately
IF EXISTS (SELECT name FROM sysobjects WHERE name = N'usp_ReIndex' AND type = N'P')
Begin
        DROP PROCEDURE usp_ReIndex
        Print 'Procedure Dropped'
End
GO
-------------------------------------------------------------------------------
-- Object: usp_ReIndex
-- Desc: Use this Procedure to ReIndex the MM SQL REpository whenever a
-- Merge/Save of big model is done to MM
-- Change History:
-- Name Date Reason
-------------------------------------------------------------------------------
CREATE PROCEDURE usp_ReIndex AS
    Declare
            @Cmd varchar(2000),
            @Name Sysname(100) ,
            @output_str varchar( 255 )
    DECLARE tmp_Reindex CURSOR FOR
            SELECT name
            FROM sysobjects
            WHERE type = 'U'
    OPEN tmp_Reindex
      FETCH tmp_Reindex INTO @Name
      WHILE ( @@sqlstatus = 0 )
      BEGIN
            SELECT @output_str = 'Processing Index for Table '+@Name
            Print @output_str
            --Set @Cmd = 'DBCC REINDEX (' + @Name + ')'
            --Exec (@Cmd)
      DBCC REINDEX (@Name )
      If @@Error <> 0
    Begin
            SELECT @output_str = 'Processing Index for Table1 '+@Name
            Print @output_str
      end
      FETCH tmp_Reindex INTO @Name
    END
    Close tmp_Reindex
go
EXEC usp_ReIndex
```

Troubleshooting the Reindexing Script

If your query tool responds:

```
no rows selected
not spooling currently
```

it means you do **not** have the correct name for the mart schema-owner. You must get the correct user name, replace 'MODELMART' with the user name, and re-run the script.

Run the Database Statistics

Database statistics show the distribution of the data in the database, and how the data is stored. When a database executes a query, it uses an optimizer to determine the best path to access the data. The optimizer relies on execution plans that specify the order in which the database accesses the tables and the exact steps used to pull the data. The database bases the creation of the execution plans on the database statistics.

As the data grows and changes, the statistics quickly become outdated and no longer reflect the true condition of the database. As a result, the execution plans no longer apply and the optimizer makes poor decisions when processing queries. You should run statistics regularly to ensure the maximum performance of your database and, therefore, your mart. We recommend that you run the database statistics nightly, and create an automated batch job to simplify the process.

Example: Oracle Statistics Commands

Note: In the following examples, the schema owner name is 'MODELMART'

To collect statistics for the mart (does not include the index statistics):

```
EXEC DBMS_STATS.gather_schema_stats ('MODELMART',DBMS_STATS.AUTO_SAMPLE_SIZE);
```

To collect statistics for the mart (includes the index statistics):

```
EXEC DBMS_STATS.gather_schema_stats (ownname => 'MODELMART',
cascade =>true, estimate_percent => dbms_stats.auto_sample_size);
```

To delete the statistics:

```
EXEC DBMS_STATS.delete_schema_stats ('MODELMART');
```

Example: Microsoft SQL Server Statistics Script

```
update statistics m7Access with sample 50 percent
update statistics m7Action with sample 50 percent
update statistics m7ActionGrouping with sample 50 percent
update statistics m7Application with sample 50 percent
update statistics m7BranchLog with sample 50 percent
update statistics m7Class with sample 50 percent
update statistics m7Control with sample 50 percent
update statistics m7ControlCache with sample 50 percent
update statistics m7Deleted with sample 50 percent
update statistics m7Library with sample 50 percent
update statistics m7License with sample 50 percent
update statistics m7Lock with sample 50 percent
update statistics m7LockType with sample 50 percent
update statistics m7Master with sample 50 percent
update statistics m7Object with sample 50 percent
update statistics m7ObjectProperty with sample 50 percent
update statistics m7OldVersions with sample 50 percent
update statistics m7Permission with sample 50 percent
update statistics m7PermissionProfile with sample 50 percent
update statistics m7PermissionType with sample 50 percent
update statistics m7ProfileAssignment with sample 50 percent
update statistics m7Property with sample 50 percent
update statistics m7Query with sample 50 percent
update statistics m7Session with sample 50 percent
update statistics m7User with sample 50 percent
```

Index