

AllFusion® Model Manager

Implementation Guide

r7.2



This documentation and any related computer software help programs (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of the Documentation for their own internal use, and may make one copy of the related software as reasonably required for back-up and disaster recovery purposes, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the provisions of the license for the product are permitted to have access to such copies.

The right to print copies of the Documentation and to make a copy of the related software is limited to the period during which the applicable license for the product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

EXCEPT AS OTHERWISE STATED IN THE APPLICABLE LICENSE AGREEMENT, TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

The use of any product referenced in the Documentation is governed by the end user's applicable license agreement.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Copyright © 2006 CA. All rights reserved.

CA Product References

This document references the following CA products:

- AllFusion® Model Manager
- AllFusion® ERwin® Data Modeler
- AllFusion® Process Modeler

Contact Technical Support

For online technical assistance and a complete list of locations, primary service hours, and telephone numbers, contact Technical Support at <http://ca.com/support>.

Contents

Chapter 1: Modeling in the Multi-User Environment	9
Understanding the Client/Server Relationship	9
System Requirements Verification	10
How You Prepare Your DBMS Environment	10
Server Connection Software	11
Database Creation Requirements	11
How You Get the Database Ready for the AllFusion MM Installation	12
 Chapter 2: AllFusion MM Installation and Initialization	 13
Before You Install	13
Install the Administrative Files	13
License AllFusion Model Manager	15
How a Database is Initialized	16
Initialize the Database on a Microsoft SQL Server or Sybase DBMS	17
Check Settings in the SQL Server Client Network Utility	18
Initialize the Database on an Oracle DBMS	19
 Chapter 3: Upgrade AllFusion MM	 21
How to Upgrade AllFusion MM	21
Microsoft SQL Server or Sybase DBMS Upgrade Considerations	22
Oracle DBMS Upgrade Considerations	22
AllFusion MM 4.1.4 to AllFusion MM r7.2 Conversion	22
Software Upgrades	23
Install the Software Upgrade	24
Conversion Process	24
Property Conversion	25
Model Property Conversion	25
Conversion Methods	25
Manual Conversion	26
Automated Conversion	26
Convert an AllFusion MM 4.1.4 Mart on a Microsoft SQL Server or Sybase DBMS to r7.2	27
Convert an AllFusion MM 4.1.4 Mart on an Oracle DBMS to r7.2	28
Refresh the Database After Upgrading Your License	29

Chapter 4: Create the Database **31**

How You Create the Database on a Microsoft SQL Server DBMS	31
How You Create the Database on a Sybase DBMS.....	32
Post Database Creation Tasks for Microsoft SQL and Sybase	33
Microsoft SQL Server Connections.....	33
Microsoft SQL Server 2005 Permissions	33
Sybase Connections.....	34
Useful Transact SQL Commands	34
How You Create the Database on an Oracle DBMS	35
Oracle Connections	39
AllFusion MM Objects Installation	39
Oracle SQL Commands	40

Chapter 5: Connect to the Database **41**

AllFusion MM Connection Prerequisites	41
Connect to the Database	42

Appendix A: Microsoft SQL and Sybase Server Tuning Recommendations **43**

Physical Tuning Parameters	43
Hardware Configurations	43
Database Placement	44
Data Redundancy	44
Server-Level Tuning Parameters	44
Named Cache Configuration	45
Database Tuning Parameters	45
Transaction Log.....	45
Threshold Procedures	46
"trunc. log on chkpt" Option	46
Database and Log Sizing	46
How to Maintain an Efficient Database	47
Nightly Maintenance Script	48
DBCC Commands	48

Appendix B: Oracle Tuning Recommendations **49**

Recommended Disk Configurations	49
Database Configuration	50
Maximum Number of Processes	50
Shared Pool	50
Buffer Cache	51
Redo Log Files	52

Increase Space in Redo Logs	53
Redo Log Buffer	55
Checkpoints	55
Storage Parameters	56
Rollback Segments	56
Set Optimal Configuration for the Rollback Logs	57
Tablespaces	59

Index	61
--------------	-----------

Chapter 1: Modeling in the Multi-User Environment

AllFusion Model Manager (AllFusion MM) coordinates the development and management of models created with AllFusion ERwin Data Modeler (AllFusion ERwin DM) and AllFusion Process Modeler (AllFusion PM).

This section contains the following topics:

[Understanding the Client/Server Relationship](#) (see page 9)

[System Requirements Verification](#) (see page 10)

[How You Prepare Your DBMS Environment](#) (see page 10)

[How You Get the Database Ready for the AllFusion MM Installation](#) (see page 12)

Understanding the Client/Server Relationship

Many security and administrative features related to AllFusion MM's multiuser capabilities and the client/server environment in which it operates are supported. For this reason, routine management of the AllFusion MM environment is usually performed by a dedicated administrator. 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. Administrators can find detailed information regarding their tasks and responsibilities in the *AllFusion Model Manager Administrator Guide*.

A client/server processing environment is shared between the user's client workstation and the server where the models are stored. Users access the stored models through one of the clients-AllFusion ERwin DM or AllFusion PM. Models are held in a database or mart, which resides on a central server, using Microsoft SQL Server, Oracle, or Sybase as the host database.

After a model is saved to a mart, the administrator can control who can work on the model and how changes can be saved to the model, making it possible for workgroups to work on large models without confusion.

You can also use AllFusion Model Navigator (AllFusion MN) as a client. Read-only access is permitted to data models and process models, so your workgroup can use AllFusion MN 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.

System Requirements Verification

Before you install the software, you must verify that your system requirements meets at least the minimum specified requirements, and that you have verified and prepared your DBMS environment for installation.

How You Prepare Your DBMS Environment

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

Install the DBMS

Install the DBMS on the server where you plan to store AllFusion MM. For more information about memory and disk space requirements, see the system requirements for the supported versions of each DBMS (Microsoft SQL Server, Sybase, and Oracle).

Create or identify the AllFusion MM mart

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

Create a user profile for each AllFusion MM user

Use the DBMS features to create a user profile for each AllFusion MM user in the database where AllFusion MM is stored. For more information, see your DBMS administrator's documentation.

Note: Each user must have Windows authentication enabled (on databases that support Windows authentication) or a valid login name on the DBMS and permission to access the database in which the AllFusion MM mart is stored.

Install the DBMS client software on each client workstation

To connect a client workstation to the DBMS server, you must install the appropriate client connection software (for example, SQL*Net) on each workstation. The connection software you use depends on your DBMS type.

Verify the client/server connection

Use a test command (for example, ping) to check 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 check the connection. For more information about testing the client/server connection, see your DBMS documentation.

Install a client on each client workstation

Install the client software using the CD-ROM that shipped with each client (AllFusion ERwin DM or AllFusion PM). Then, any user with an AllFusion MM login and password and the appropriate DBMS client connection software can connect to the mart from that client workstation.

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.

Database Creation Requirements

You must be the dbo in the target database on the Microsoft SQL or Sybase server or the DBA for the Oracle server to create the AllFusion MM mart.

Note: The DBA role is necessary only during the software installation or upgrade. It is not necessary after you create the mart.

How You Get the Database Ready for the AllFusion MM Installation

Before you can install AllFusion MM in your DBMS environment you must do the following:

Note: You must be the database owner (dbo for Microsoft SQL Server or Sybase) or the schema owner (Oracle) of the database to create or delete an AllFusion MM mart.

- Install the administrative files
- Initialize the database on your DBMS server
- Upgrade your license agreement permitting more users to access AllFusion MM
- Upgrade your current version of software when a newer version is available

Chapter 2: AllFusion MM Installation and Initialization

This section contains the following topics:

[Before You Install](#) (see page 13)

[Install the Administrative Files](#) (see page 13)

[License AllFusion Model Manager](#) (see page 15)

[How a Database is Initialized](#) (see page 16)

Before You Install

Review the hardware and software requirements and the list of supported databases in the readme file before you install the program.

If you installed an earlier version of AllFusion MM (pre-r7) it is not necessary to remove that version. Since this version 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.

Install the Administrative Files

The installation of the administrative files is the same for each DBMS supported (Microsoft SQL Server, Sybase, or Oracle). You can initialize the mart at the end of the installation process, or you can initialize the mart at a later time.

To install the administrative files

1. Insert the installation CD in the CD drive.

If Autorun is enabled, you will be prompted to install. If autorun is disabled, click on the CD and double-click on Setup.exe.

The Welcome dialog opens.

2. Click Next to continue.

A License Agreement opens for you to review.

3. Select the option to accept the terms as described in the License Agreement and continue. Follow the instructions provided by the installation wizard. The wizard asks a series of questions you must answer, including:
 - Customer Information
 - Destination folder for the AllFusion Model Manager files
4. Select Complete or Custom Setup Type.

Complete

Installs all program features.

Custom

Installs selected features. A new window opens with options to select specific features for installation. You can also evaluate the space needed to install each feature, and change the destination folder for the files.

Click Next to continue and then click Install to start the installation process.

The program files are copied.

5. When the install completes, a final screen displays with the following two options:

Launch CA AllFusion Model Manager

Clear the check box and click Finish to initialize the mart later. You can begin initialization from the Start menu on your desktop when you want to initialize the mart.

Show the readme file

Clear the check box if you do not wish to review the readme file at this time.

Both check boxes are selected by default so you can review the readme file, and proceed with initializing the mart. Click Finish to exit the wizard.

The installation is complete.

Important! You need to install or upgrade the license key before you initialize the mart. Follow the instructions to install the license key before you launch the application in this section.

License AllFusion Model Manager

If you have not already licensed AllFusion MM, you can license the program at this time. Locate the 25-character License Key that accompanies the product CD. Typically, this key is found on the back of the CD sleeve. You need this License Key to complete the license process.

To license AllFusion MM

1. Select a component in the Component List in the License Verification dialog and select an option:

Use Trial Version

Install the Try Buy license. The product runs for 30 days before requiring a license.

Use License Key

Enter a license key at this time.

2. Enter the License Key in the License Key field for the first component listed in the License Verification dialog, and click Add.

The information you entered appears in the License Keys field.

Note: If you enter an Upgrade license key in the License Key field, and the component is not currently licensed on your system, the Registration ID field appears, allowing you to enter the Registration ID for your upgraded component.

3. Select the next component you are licensing from the Component List, enter the License Key for this component in the License Key fields and click Add.

Repeat these steps until you have entered License Keys for all of the components you are licensing.

4. Click Continue.

The installation process continues.

How a Database is Initialized

After you create a database and install the administrative files, you must initialize the database to create the tables and install the stored procedures that manage client workgroup models. You cannot use AllFusion MM until you initialize it.

Important! You need to install or upgrade the license key before you initialize the database. Follow the instructions in the AllFusion Model Manager Implementation Guide to install the license key before you launch the application.

To initialize a database, you must connect to the database by either of the following methods:

- At the end of the installation process when you select the Launch Application check box and click Finish.
- When you choose AllFusion Model Manager in the AllFusion Model Manager program group in the Start menu.

You must be logged on to the DBMS as the dbo (Microsoft SQL Server or Sybase) or the schema owner (Oracle) of the database that contains AllFusion MM.

The initialization process does the following:

- Creates the necessary tables and installs the stored procedures and queries that manage client workgroup models in the database.
- Prepares the database for use with both AllFusion ERwin DM and AllFusion PM.

The complete initialization steps are slightly different depending upon the DBMS. Follow the appropriate section for your DBMS.

Initialize the Database on a Microsoft SQL Server or Sybase DBMS

Before you can begin working, you must initialize the database, which prepares it for use with both AllFusion ERwin DM and AllFusion PM. You must be logged on to the DBMS as the dbo of the database that contains AllFusion MM.

Important! The AllFusion MM mart and the AllFusion MM Control Tables (m7Master and m7License) must be installed on the same database.

To initialize the database on a Microsoft SQL Server or Sybase DBMS

1. Click Start, Programs, CA, AllFusion, Model Manager r7, Model Manager r7.
The Connection Manager opens.
2. Complete the following information in the Connection Manager.

Database

Select the DBMS version (for example, Microsoft SQL Server 2000/2005 - using db-lib).

Authentication

Windows Authentication

Use if you use Windows usernames and passwords to secure database access and are currently logged onto your machine as the user who is the dbo of the database. For further information about Windows Authentication in Oracle, please consult your Oracle documentation or DBA.

Database Authentication

Use to enter the login name and password for the dbo of the database.

Server Parameters

Enter the server name (for example, sqlmart).

Database Parameters

Enter the name of the master database. If you leave Master Database blank, the Select Master Database dialog opens for you to select the master database from a list.

You can select a database connection from the Recent Connections panel to automatically populate the Database, and Parameters previously used.

Click Connect.

The AllFusion MM Manager opens.

3. Click Truncate Log Before Any Action if you want 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 may cause the database to hang.

4. Click Create in the AllFusion MM Manager dialog, after verifying the master database name in the Database field to create the tables and stored procedures in the selected database.

The Security Manager opens.

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

In the Security Manager dialog, your user name is automatically added to the Administrator profile in the Security Profile list.

5. Use the controls in the Security Manager dialog to assign additional security profiles. You can also do this later from AllFusion ERwin DM or AllFusion PM. When you are done assigning profiles, click OK.

The Setup Complete message displays.

6. Click OK to return to Windows.

The database is initialized.

Check Settings in the SQL Server Client Network Utility

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

To verify that the settings in your configuration are correct

1. Click Programs, Microsoft SQL Server, Client Network Utility from 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

Click OK.

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

Initialize the Database on an Oracle DBMS

Before you can begin working, you must initialize the database, which prepares it for use with both AllFusion ERwin DM and AllFusion PM. You must be logged on to the DBMS as the schema owner of the database that contains AllFusion MM.

To initialize the database on an Oracle DBMS

1. Click Start, Programs, CA, AllFusion, Model Manager r7, Model Manager r7.
The Connection Manager opens.
2. Complete the following information in the Connection Manager.

Database

Select the Oracle DBMS version (for example Oracle 9i).

Authentication

Windows Authentication

Select if you use Windows usernames and passwords to secure database access and are currently logged onto your machine as the user who is the dbo of the database containing AllFusion MM. For further information about Windows Authentication in Oracle, please consult your Oracle documentation or DBA.

Database Authentication

Select to enter the login name and password for the dbo of the database containing AllFusion MM.

Connection String

Enter the server name (for example, Mymart).

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

Click Connect.

The AllFusion MM Manager opens.

3. Select the following from in the AllFusion MM Manager dialog:

AllFusion MM Role

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

Table Tablespace

Select where you want to store the AllFusion MM tables.

Index Tablespace

Select where you want to store the AllFusion MM indexes.

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

Click Create.

The AllFusion MM objects are created and the Security Manager dialog opens.

Important! If you do not see the Security Manager dialog, it may 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.

4. Use the controls in the Security Manager dialog to assign additional security profiles.

Note: You can also do this later from AllFusion ERwin DM or AllFusion PM.

When you are done assigning profiles, click OK.

The Setup Complete message opens.

5. Click OK.

The database is initialized.

Chapter 3: Upgrade AllFusion MM

This section contains the following topics:

[How to Upgrade AllFusion MM](#) (see page 21)

[Software Upgrades](#) (see page 23)

[Install the Software Upgrade](#) (see page 24)

[Conversion Process](#) (see page 24)

[Conversion Methods](#) (see page 25)

[Convert an AllFusion MM 4.1.4 Mart on a Microsoft SQL Server or Sybase DBMS to r7.2](#) (see page 27)

[Convert an AllFusion MM 4.1.4 Mart on an Oracle DBMS to r7.2](#) (see page 28)

[Refresh the Database After Upgrading Your License](#) (see page 29)

How to Upgrade AllFusion MM

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 may exist, and will 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 prior to version 4.1.4 in your mart using an AllFusion 4.1.4 client before you convert to AllFusion MM r7.2.
- 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 AllFusion ERwin DM and AllFusion PM 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 the AllFusion MM.
- Turn on AUTOEXTEND to automatically extend the AllFusion MM datafile 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 (SQL2000 DBMSs only).
- Review the Readme 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 512 MB RAM (1 GB recommended) 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 AllFusion MM.

Microsoft SQL Server or Sybase DBMS Upgrade Considerations

On a Microsoft SQL Server or Sybase DBMS, if you are updating from a previous AllFusion MM version and the system logs are full, you may not be able to upgrade your previous AllFusion MM 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 AllFusion MM datafile when full.

AllFusion MM 4.1.4 to AllFusion MM r7.2 Conversion

Verify that the AllFusion MM mart is version 4.1.4 (Build 3643 or later). If it is a version earlier than 4.1.4, you need to upgrade to 4.1.4 before you can upgrade to r7.2. We recommend you upgrade to the latest Service Pack for AllFusion MM Version 4.1.4. You must open and save your models older than AllFusion MM version 4.1.4 in AllFusion MM Version 4.1.4 before you can upgrade to AllFusion MM r7.2. See <http://ca.com/supportconnect/> for details on upgrading to AllFusion MM Version 4.1.4.

Software Upgrades

CA periodically releases new versions of the software and supported client software products. 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 as the dbo (for Microsoft SQL Server or Sybase) or the schema owner (for Oracle).

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

Note: For some software upgrades, a new license key 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.

Install the Software Upgrade

To upgrade the software to a new version you must log on to your DBMS as either the dbo (Microsoft SQL Server or Sybase) or schema owner (Oracle) of the database that contains the AllFusion MM mart.

Note: For some software upgrades, a new license key is required. Locate the 25-character License Key that accompanies the product CD. Typically, this key is found on the back of the CD sleeve.

To install the software upgrade

1. Close all client connections and insert the CD in your CD-ROM drive.
 - If your autorun function is enabled, the Setup program starts automatically.
 - If your autorun function is not enabled, follow these steps to start the Setup program:
 - a. Click Start on the taskbar, and then select Run.
The Run dialog opens.
 - b. Type d:\setup in the Run dialog, where d is your CD-ROM drive and click OK.
The Setup program starts.
2. Select the Launch Application check box when the InstallShield Wizard Complete dialog opens, and then click Finish.
The software upgrade installation completes and the Connection Manager opens.

Conversion Process

The conversion systematically converts a mart, including all AllFusion MM properties and all client AllFusion ERwin DM and AllFusion PM 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 AllFusion MM 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 AllFusion ERwin DM and AllFusion PM model properties including:

Model data

AllFusion ERwin DM and AllFusion PM models in the mart are converted. The converted models are the same as if opening an .er1 file with the upgraded client (r7.2) and saving the model to AllFusion MM r7.2.

All model objects

AllFusion ERwin DM and AllFusion PM accept 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

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

Automated Conversion

Install the new software version and initialize the mart in the same location as the old mart. Choose Convert to automatically convert all existing AllFusion ERwin DM and AllFusion PM 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. The following are some advantages to the manual method:

- Selectively choose the models you want to move, and leave those that you do not.
- “Clean up” your mart if there are many models or versions which are no longer in use, and which would extend conversion time.
- Run both AllFusion Model Manager versions simultaneously, and move incrementally, model by model, to the new mart.
- Host your new mart on a different server or database platform than your old mart version.

The disadvantages are that the version history is lost on models checked into the new mart. Being a manual process, it is time consuming and labor intensive.

Automated Conversion

The automated conversion uses the Update option on the AllFusion MM Manager dialog. Your existing AllFusion ERwin DM and AllFusion PM 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 and all models retain their full history. 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 may require you to move your old version of the mart, if you cannot create additional space.

Convert an AllFusion MM 4.1.4 Mart on a Microsoft SQL Server or Sybase DBMS to r7.2

All the existing AllFusion ERwin DM and AllFusion PM models in the AllFusion MM 4.1.4 mart are initialized and prepared for the r7 internal structure. The model is not fully converted until you open and save the model in AllFusion MM r7.2 with AllFusion ERwin DM or AllFusion PM for the first time. It may take a few minutes longer to open the model the first time to complete the process.

To convert an AllFusion MM 4.1.4. mart to an AllFusion MM r7.2 mart

1. Log on to your DBMS machine that contains AllFusion MM as the dbo, and click Start, Programs, CA, AllFusion, Model Manager r7, Model Manager r7.

The Connection Manager opens.

2. Complete the information in the Connection Manager and click Connect.

The AllFusion MM Manager dialog opens, indicating that an AllFusion MM mart exists and the status of that mart.

3. Click Truncate Log Before Any Action if you want 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 may cause the mart upgrade to hang. You cannot remove the transaction log after conversion begins.

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

Create

Creates an empty mart in the same database as the current AllFusion MM, and then opens and saves all existing models to convert the data.

Convert

Creates a new mart with the current version and copies the data from the old mart into the new mart.

The AllFusion MM Manager displays conversion progress messages.

5. Click OK on the Upgrade has Successfully Completed message dialog.

The status is shown.

6. Click Close to exit out of the dialog.

The administrative setup is complete.

7. Click OK on the Setup Complete dialog.

The conversion is complete.

The same user profile assignments exist as before the conversion. If you need to make changes to user profiles, you can access the Security Manager in the Services menu or click the User Permissions icon on the AllFusion MM Toolbar.

Convert an AllFusion MM 4.1.4 Mart on an Oracle DBMS to r7.2

All the existing AllFusion ERwin DM and AllFusion PM models in AllFusion MM 4.1.4 are initialized and prepared for the r7.2 internal structure. The model is not fully converted until you open and save the model in AllFusion MM r7.2 with AllFusion ERwin DM or AllFusion PM for the first time. It may take a few minutes longer to open the model the first time to complete the process.

To convert your AllFusion MM 4.1.4. mart to a AllFusion MM r7.2 mart

1. Log on to your DBMS machine that contains AllFusion MM as the schema owner and click Start, Programs, CA, AllFusion, AllFusion Model Manager r7, Model Manager r7.

The Connection Manager opens.

2. Complete the information in the Connection Manager and click Connect.

The AllFusion MM Manager dialog opens, indicating that an AllFusion MM mart exists and the status of that mart.

3. Select the AllFusion MM Role (MMUSER) used to initialize AllFusion MM 4.1.4 in the Role drop down list, and the tablespace that contains the AllFusion MM 4.1.4 mart you want to convert.

The dropdown box is not active if there is only one tablespace.

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

Create

Creates an empty mart in the same database as the current AllFusion MM, and then opens and saves all existing models to convert the data.

Convert

Creates a new mart with the current version and copies the data from the old mart into the new mart.

The AllFusion MM Manager displays conversion progress messages.

5. Click OK on the Upgrade has Successfully Completed message dialog.

The status is shown.

6. Click Close to exit out of the dialog.

The administrative setup is complete.

7. Click OK on the Setup Complete dialog.

The conversion is completed.

The same user profile assignments exist as before the conversion. If you need to make changes to user profiles, you can access the Security Manager in the Services menu or click the User Permissions icon on the AllFusion MM toolbar.

Refresh the Database After Upgrading Your License

To upgrade your license, you must be assigned to the Administrator security profile and you must be logged on as the dbo (for Microsoft SQL Server or Sybase) or the schema owner (for Oracle). After you upgrade the license file, you must refresh AllFusion MM.

To refresh AllFusion MM

1. Click Start on the taskbar, and then select Programs, CA, AllFusion, Model Manager r7, Model Manager r7.

The Connection Manager dialog opens.

2. Connect to the database where you upgraded the license.

The connection is made.

3. Click Close.

The database refresh is automatic.

Note: If you upgrade your user license with the software upgrade, the Setup program automatically detects your new license key and a message displays confirming the license upgrade. You should install the new license file before you install the software upgrade.

Chapter 4: Create the Database

This section contains the following topics:

[How You Create the Database on a Microsoft SQL Server DBMS](#) (see page 31)

[How You Create the Database on a Sybase DBMS](#) (see page 32)

[Post Database Creation Tasks for Microsoft SQL and Sybase](#) (see page 33)

[How You Create the Database on an Oracle DBMS](#) (see page 35)

How You Create the Database on a Microsoft SQL Server DBMS

The following tasks must be performed by the DBA and system administrator responsible for installing AllFusion MM 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 AllFusion MM 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 allowed to connect to the database server; a user gives a login the right to use a particular database. If you use Windows security, you must add Windows users to the database.

3. Check 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.

How You Create the Database on a Sybase DBMS

The following tasks must be performed by the DBA and system administrator responsible for installing the software on a Sybase database management system. Information provided in this section includes installation and Sybase connectivity software requirements.

1. Use a graphical user interface (GUI) tool, ISQL (all versions), SQL Administrator, Sybase Central Java, or Enterprise Manager to create the AllFusion MM 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 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. You should also locate the data device on a different disk (and disk controller) than the transaction log.
- For optimum performance, ensure 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 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 allowed to connect to the database server; a user gives a login the right to use a particular database.

3. Check 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 AllFusion MM client invokes these stored procedures to control changes to the data in the database.

4. Check 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.

5. Check memory allocated to the database server.

Allocate at least 32 MB of RAM to the database server. The amount of RAM allocated should be half of the available RAM on the server machine.

Post Database Creation Tasks for Microsoft SQL and Sybase

Run the Administrative Setup program located on the CD-ROM. The Setup program prompts you for an installation directory. By default this directory is C:\Program Files\CA\AllFusion Model Manager r7, 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 tips on how to determine if your installation computer can connect to the target server, see Microsoft SQL Server Connections or Sybase Connections.

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.'

Microsoft SQL Server Connections

AllFusion MM uses DB_LIBRARY to connect to Microsoft SQL server.

Microsoft SQL Server 2005 Permissions

For SQL Server 2000, you only need to have public assigned in order to save to the mart. However, when the repository is on an SQL Server 2005 instance, you need to have the bulkadmin permission designated as well. The ability to do bulk inserts (which was permitted by public, previously) is no longer part of the public permission. You must explicitly define this permission or when you attempt to save a model to a new AllFusion Model Manager instance created using an SQL Server 2005 database, an error "You do not have permission to use the bulk load statement." is returned.

Sybase Connections

AIIFusion MM uses CT_LIBRARY to connect to Sybase.

Useful Transact SQL Commands

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

Example: Transact SQL DISK INIT Command

The following is an example of how to 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: Transact SQL CREATE DATABASE Command

The following is an example of how to 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: Transact SQL `sp_addlogin` and `sp_adduser` Commands

The following is an example of how to 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]]]
```

and

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

Once you have executed 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]
```

where `login_id` is the login ID of the new database owner.

How You Create the Database on an Oracle DBMS

The following tasks must be performed by the DBA and system administrator responsible for installing the software on an Oracle database management system. Information provided in this section includes installation and Oracle connectivity software requirements.

These steps may be performed using graphical tools but you can also use SQL*Plus (all versions), SQL*DBA (either command line or full screen mode), or the Oracle Enterprise Management Console that ships with Oracle Version 8.0 or later. 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 will often need 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 tablespace.

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 on syntax and options regarding tablespace creation, see the appropriate Oracle documentation.

6. Create an Oracle user with DBA privileges to be used by the AllFusion MM 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 an AllFusion MM Installer role.

The following example is the role required by the Oracle user installing AllFusion MM.

```
CREATE ROLE MMINSTALL;
```

8. Grant Oracle privileges to the Installer role.

The following example shows the Oracle privileges that the AllFusion MM 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 an AllFusion MM User role.

For example:

```
CREATE ROLE MMUSER;
```

When you select this role as the AllFusion MM 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 AIfusion MM 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 AIfusion MM User role to the AIfusion MM Installer role.

For example:

```
grant MMUSER to MMINSTALL;
```

13. Grant the AIfusion MM Installer role to the AIfusion MM Installer user. You must also grant DBA privileges to the user you designate as the AIfusion MM 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, choose the Initialize AIfusion MM option and connect to Oracle as the AIfusion MM Installer user (for example, STEVE).

15. Select the AIfusion MM tablespaces and AIfusion MM User role.

When prompted for tablespace and role information, choose the AIfusion MM data tablespace, the AIfusion MM index tablespace, and the AIfusion MM User role (the role created in Step 9).

16. Add the AIfusion MM Installer user as the AIfusion MM 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 AIfusion MM Installer role and grant the AIfusion MM User role to the AIfusion MM Installer user.

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 machines. These files can be generated by the Oracle networking tools, but can also be created using a text editor. Since Oracle will 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 machine and the Oracle SIDs for the databases known on that machine

AllFusion MM Objects Installation

Installing the software on an Oracle DBMS creates the following database objects:

- Tables
- Indexes
- Stored procedures
- Public synonyms

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 TABLESPACE Command-- Oracle Platform

To create a tablespace, use 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 USER Command-- Oracle Platform

To create an Oracle user, use the CREATE USER command:

```
CREATE USER mm_user_1 /* The user id name */
IDENTIFIED 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 Command-- Oracle Platform

To grant a user privileges, use the GRANT command:

```
GRANT role_or_privilege_name
TO user_or_role
;
```


Chapter 5: Connect to the Database

This section contains the following topics:

[AllFusion MM Connection Prerequisites](#) (see page 41)

[Connect to the Database](#) (see page 42)

AllFusion MM Connection Prerequisites

Users connect to AllFusion MM on the server through the client software contained in AllFusion ERwin DM, AllFusion PM, or AllFusion MN. You must perform the following tasks before you can connect to AllFusion MM through the client:

- The administrator of the relational database management system (DBMS) must give users permission to access the DBMS, on which the AllFusion MM database is set up.
- The AllFusion MM administrator must install the administrative files 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.
- 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 AllFusion MM client software. You can find the requirements in the documentation provided with these products.
- 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.

See the Administrator Guide for further instructions on these administrator tasks.

Connect to the Database

You can establish a connection to a target database or the AllFusion MM mart. You need to connect to AllFusion MM to share and access models with other members of the workgroup.

To connect to the database

1. Click Connection from the Services menu.

The Connection Manager opens.

2. Complete the following information in the Connection Manager.

Database

Select the DBMS version (for example, Microsoft SQL Server 2000/2005 - using db-lib).

Authentication

Windows Authentication

Use if you use Windows usernames and passwords to secure database access and are currently logged onto your machine as the user who is the dbo of the database. For further information about Windows Authentication in Oracle, please consult your Oracle documentation or DBA.

Database Authentication

Use to enter the login name and password for the dbo of the database.

Server Parameters

Enter the server name (for example, sqlmart).

Database Parameters

Enter the name of the master database. If you leave Master Database blank, the Select Master Database dialog opens for you to select the master database from a list.

You can select a database connection from the Recent Connections panel to automatically populate the Database, and Parameters previously used.

Click Connect.

The dialog closes. You are now connected to the mart and can begin a session.

Important: If you have an existing version of software and you want to enable Windows authentication for the first time, be sure to connect using database authentication. You can then enable Windows authentication for designated users.

Appendix A: Microsoft SQL and Sybase Server Tuning Recommendations

The AllFusion MM 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.

This section contains the following topics:

[Physical Tuning Parameters](#) (see page 43)

[Server-Level Tuning Parameters](#) (see page 44)

[Named Cache Configuration](#) (see page 45)

[Database Tuning Parameters](#) (see page 45)

[Database and Log Sizing](#) (see page 46)

[How to Maintain an Efficient Database](#) (see page 47)

Physical Tuning Parameters

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

Hardware Configurations

Consider implementing AllFusion MM 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 machine. The database's 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 AllFusion MM on server machines that have at least 512 MB of RAM.

Database Placement

AllFusion MM 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 the transaction log is mirrored on a separate drive to maximize recoverability of the database. If the uptime of the database is considered critical, mirror both the database and the transaction logs; you may also want to mirror the other Microsoft SQL Server files. Consider OS mirroring to speed the mirroring process and make it easier to manage.

Server-Level Tuning Parameters

Server-level tuning task recommendations are:

- 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 since AllFusion MM uses many stored procedures.
- Configure two additional user connections for each simultaneous AllFusion MM user. Two connections per session are used.

The following shows the recommended configuration parameters for a Microsoft SQL Server running AllFusion MM:

Parameter Type	Recommended Value	Notes
memory (Sybase 11: "total memory")	512 MB minimum Note: 1 Gig is recommended	More memory implies less physical I/O
procedural cache	30 percent	AllFusion MM is stored procedure intensive
user connections	2 times the number of concurrent sessions	Add this value to existing user connection values

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, and can be backed up independently of the database by issuing a `DUMP TRANSACTION` command. Microsoft SQL Server keeps a transaction log for each database.

Note: A side effect of the `DUMP TRANSACTION` command is that it clears the inactive transactions from the log to reduce its size.

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: Since 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 will be 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 AllFusion MM 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 AllFusion MM marts:

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

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 refer to the appropriate Oracle documentation.

If the Oracle server is not dedicated to AllFusion MM, 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.

This section contains the following topics:

[Recommended Disk Configurations](#) (see page 49)

[Database Configuration](#) (see page 50)

[Storage Parameters](#) (see page 56)

Recommended Disk Configurations

The layout of Oracle database files on physical disk drives plays a major role in the performance of the database. When used with AllFusion MM, 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 both 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 may prevent some users from accessing the server during peak usage times. Setting the number of processes at a higher value may waste a small amount of system memory resources.

Shared Pool

The shared pool is an area of Oracle's 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's 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 may degrade performance, forcing Oracle to do disk I/O to retrieve objects that cannot fit into 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 rather than 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 into 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 may 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 into 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 that happens, 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. Since AllFusion MM 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 may 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 AllFusion MM, 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 our 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: In the previous table, “# of transactions” is the maximum number of simultaneous AllFusion MM 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. Login 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 into 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 Gig
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

Index

B

Buffer cache • 51

C

Checkpoints • 55

Client/Server

understanding • 9

Create

an Oracle user • 40

database • 31, 34

tablespace • 40

D

Database

configuration • 50

connect • 41

convert • 24, 25, 26, 27, 28

create • 11, 12, 31

database and log sizing • 46

initialize • 13, 17, 19

maintenance • 47

options • 46

placement • 44

post creation tasks • 33

tuning parameters • 45

upgrade recommendations • 21

I

Install

administrative files • 13

AllFusion MM Objects • 39

and initialize AllFusion MM • 13

software upgrade • 24

M

Microsoft SQL Server

connections • 33

conversion • 27

create • 31

database creation requirements • 11

database placement • 44

initialize • 17

maintenance • 47

server tuning recommendations • 43

settings • 18

upgrade recommendations • 22

Modeling

workgroup • 9

O

Oracle Server

connections • 39

conversion • 28

create a user • 40

create the database • 35

initialize • 19

Oracle SQL commands • 40

Transact SQL commands • 34

tuning recommendations • 49

upgrade recommendations • 22

R

Recommendations

disk • 49

index • 60

table size • 60

Requirements

system • 10, 11

S

Sessions

connect to the database • 42

Sybase Server

connections • 34

conversion • 27

create database • 32

initialize • 17

server tuning recommendations • 43

upgrade recommendations • 22

T

Tablespaces • 59

Transaction log • 45

U

Upgrade

AllFusion MM • 21

recommendations • 21

software • 23

