Application Discovery Manager
Repository Reference Guide

vCenter Application Discovery Manager 6.2.2

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see http://www.vmware.com/support/pubs.
You can find the most up-to-date technical documentation on the VMware Web site at:
http://www.vmware.com/support/
The VMware Web site also provides the latest product updates.
If you have comments about this documentation, submit your feedback to:
docfeedback@vmware.com

Copyright © 2012 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at http://www.vmware.com/go/patents.
VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.
# Contents

About This Book 5

1 Introduction 7
   Overview 7
   How Data Is Collected 7
   Synchronizing the External Database 8

2 Set Up the External Reports Scheme Configuration Policy 11
   Set Up and Monitor the External Reports Scheme Configuration Policy 11
     Creating the External Reports Scheme Configuration Policy 11
   Check the Status of the Synchronization Process 13

3 External Database Setup 15
   External Database System Requirements 15
     Minimum Hardware Requirements 15
     Software Requirements 15
   Installing and Configuring the External Database for Oracle 16
     Install the Client-Side Script 16
     Create the External Reporting Database User 17
     Create the adm_olap Tables 17
     Configure the Synchronization Process 18
   Installing and Configuring the External Database for MS SQL 20
     Install the Client-Side Scripts 20
     Install the External Database for MS SQL Server 20
     Configure the Synchronization Process 21
   Manually Loading Data to the External Database 23
   Checking the Status of the External Database Load Process 23
   Upgrading the External Database 24
     Upgrading for MS SQL 24
     Upgrading for Oracle 25

4 ADM ERD Schema 27
   Overview of External Database Tables 28
     Database Structure and Dependencies 28
   Address 29
   Change_Log 29
   CPU 29
   Database 30
   Dependency 30
   Group_Membership 31
   Installed_Software 32
   IP_Route 32
   Java_EE 33
   Network_Element 33
   Network_Interface 34
Port 35
Property 35
Service 36
Storage_Device 36
Switch_Address_Forwarding 37
Time 37
Usage_Metric 38
Usage_Metric_Baseline 38
Virtual_System_Settings 38

5 ADM Custom Reports 41
Creating Custom Reports 41
Host Inventory 41
Connection Inventory 41
Most Used Services 41
Application Dependency 42

Index 43
The *VMware vCenter™ Application Discovery Manager Repository Reference Guide* provides information about setting up and configuring the Application Discovery Manager (ADM) external repository. It describes the table structure and dependencies within the ADM external database and also illustrates the tables. To help you with the custom reports, the book provides examples of reports that can be generated from the ADM external database.

**Intended Audience**

This document is part of the VMware vCenter Application Discovery Manager documentation set, and is intended for use by corporate Information Technology (IT) personnel who need to monitor enterprise applications and resources and make decisions about acquiring, allocating, and modifying these resources. Readers of this guide are expected to be familiar with these topics:

- Oracle database
- SQL

**VMware Technical Publications Glossary**

*VMware®* Technical Publications provides a glossary of terms that might be unfamiliar to you. For definitions of terms as they are used in VMware technical documentation go to [http://www.vmware.com/support/pubs](http://www.vmware.com/support/pubs).

**Document Feedback**

VMware welcomes your suggestions for improving our documentation. If you have comments, send your feedback to docfeedback@vmware.com.

**Technical Support and Education Resources**

The following sections describe the technical support resources available to you. To access the current version of this book and other books, go to [http://www.vmware.com/support/pubs](http://www.vmware.com/support/pubs).

**Online and Telephone Support**

To use online support to submit technical support requests, view your product and contract information, and register your products, go to [http://www.vmware.com/support](http://www.vmware.com/support).

Customers with appropriate support contracts should use telephone support for the fastest response on priority 1 issues. Go to [http://www.vmware.com/support/phone_support](http://www.vmware.com/support/phone_support).

**Support Offerings**

To find out how VMware support offerings can help meet your business needs, go to [http://www.vmware.com/support/services](http://www.vmware.com/support/services).
**VMware Professional Services**

VMware Education Services courses offer extensive hands-on labs, case study examples, and course materials designed to be used as on-the-job reference tools. Courses are available onsite, in the classroom, and live online. For onsite pilot programs and implementation best practices, VMware Consulting Services provides offerings to help you assess, plan, build, and manage your virtual environment. To access information about education classes, certification programs, and consulting services, go to [http://www.vmware.com/services](http://www.vmware.com/services).
This chapter introduces the VMware vCenter™ Application Discovery Manager (ADM) repository and provides an overview that includes descriptions of data collection, table types, and key fields.

This chapter includes the following topics:
- “Overview” on page 7
- “Synchronizing the External Database” on page 8

Overview

The ADM provides standard reports based on the data collected and stored in the ADM Repository. The schema of the new ADM External Database is simple to understand and allows you to write reporting queries over it without disrupting the performance of the other ADM services.

How Data Is Collected

The ADM collects, manages, and stores data through the components listed in Table 1-1.

Table 1-1. ADM Components

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Discovery-Unix</td>
<td>Collects data from the configuration objects in your data center. The following discovery types apply:</td>
</tr>
<tr>
<td></td>
<td>- IP discovery: Detects hosts or other configuration items with a specific IP address when passive discovery fails to discover them.</td>
</tr>
<tr>
<td></td>
<td>- Detail discovery: Extends the information obtained using passive and IP discovery. It uses common network protocols to remotely query servers in the managed network and obtains supplementary information about network hosts that is added to the database.</td>
</tr>
<tr>
<td>Active Discovery-Windows</td>
<td>A discovery engine that uses WMI based discovery policies for performing active discovery on Windows machines.</td>
</tr>
<tr>
<td>Passive Discovery</td>
<td>Passively observes the network traffic by performing a deep-packet analysis to discover applications and component relationships in physical and virtual environments. Passive discovery also allows you to do the following:</td>
</tr>
<tr>
<td></td>
<td>- Map dependencies.</td>
</tr>
<tr>
<td></td>
<td>- Count the activity of these dependencies.</td>
</tr>
<tr>
<td></td>
<td>- Identify services.</td>
</tr>
<tr>
<td>Aggregator</td>
<td>Receives data from the discovery components and reconciles the data before transferring it to the database component. The aggregator also provides the user interface for using ADM and is the integration point for various integrations, for example, ERDB.</td>
</tr>
<tr>
<td>Database</td>
<td>An Oracle RDBMS used to store discovered data and ADM configuration.</td>
</tr>
</tbody>
</table>

Figure 1-1 illustrates how data is collected and stored in the ADM database.
The data stored in the ADM database cannot be customized or manipulated for creating custom reports. Therefore, ADM provides functionality for exporting the data into an ADM external repository, from which custom reports are created.

The data transferred to the external repository is not real-time data. It is synchronized at regular intervals as defined in the External Reports Scheme Configuration policy.

**Synchronizing the External Database**

**Figure 1-2** illustrates how the ADM external database is synchronized with the ADM repository through the ETL (Extract-Transform-Load) process. Extraction and transformation is done through ADM and loading is done by the adm_olap component on the external database host.

The synchronization process comprises of the following steps:

1. The synchronization is triggered according to the settings in the External Reports Scheme Configuration policy.
2. ADM checks the status of the replication to ensure that the previous synchronization and that any report requests against the external database has completed.
3. ADM extracts the data files for the load process as follows:
   a. The Report Schema is extracted and translated to .csv files.
   b. The data files are compressed and placed together in a single Zip archive.
   c. The Zip file is uploaded to the assigned FTP server.
4. The system running the external database periodically checks the FTP server for data files based on the synchronization schedule defined on the external database system.
5 When a data file is found, the following process takes place on the external database system:
   a The latest Zip file is downloaded.
   b The Zip file is extracted to the decompressed .csv data files.
   c The database loading is performed using the database specific loader.

**Performance and Scalability**

The following steps indicate the amount of time required to complete each process:

- Extraction and flat file creation: This step utilizes the ADM resources to the maximum. Therefore, it is scheduled to extract the data at night or some other period of low resource consumption and to allow enough time to complete.

- Transferring the data files: This step involves uploading of the flat files from the ADM Repository appliance to the external database host. This step does not consume many ADM resources. However, the transfer time might vary greatly because of network conditions at the external database host site.

- Loading the database: This step might take few hours to complete.
The ADM Console provides the necessary functionality to:

- “Set Up and Monitor the External Reports Scheme Configuration Policy” on page 11
- “Check the Status of the Synchronization Process” on page 13

Set Up and Monitor the External Reports Scheme Configuration Policy

The External Reports Scheme Configuration policy defines the FTP site where the database files are sent and how often extraction is performed. The External Reports Scheme Configuration provides functionality for running the policy automatically at a scheduled time, or starting the synchronization now.

This policy only controls the following steps of the synchronization process:

- Database data extraction
- Database files creation
- Posting the database files to the FTP site

You must configure the external database host to retrieve the data files from the FTP site and complete the synchronization process. Chapter 3, “External Database Setup,” on page 15 provides complete details.

Creating the External Reports Scheme Configuration Policy

To set up the External Reports Scheme Configuration policy

1. Log in to the ADM Console as admin user.
2. From the ADM Console, navigate to Connectors > Reports and click Configuration in the left pane.
3. In the **Publisher** tab, enter the FTP site to which the data files are published as shown below.

![Publisher tab](image)

where:

- **Target Host**: The system hosting the FTP server.
- **Target Directory**: The directory on the FTP server host to which the files are sent.
- **User Name**: The name of a user with access to the FTP server.
- **Password**: The password associated with the user name.

**NOTE** Ensure that the user name and password are the same, as used to configure the synchronization on the external database system as described in “Installing and Configuring the External Database for Oracle” on page 16 for Oracle databases user and “Installing and Configuring the External Database for MS SQL” on page 20 for MS SQL Server users.

- **Transport Channel**: Allows user to choose between FTP and Secured FTP (SFTP) for data upload from the drop-down menu.
- **Number of Retries**: The number of times the synchronization process attempts, in case of failure.

4. Click **Test Connection** to test the connection to the FTP server based on the publishing credential you entered. If you have entered invalid credentials or if the FTP server is down, the test times out or fails.

5. Click **Create policy**. If you have previously defined the policy, click **Update**.

**Running the Synchronization Process at Scheduled Times**

**To schedule the synchronization process**

1. After creating the External Reports Scheme Configuration policy, select the **Enable Automatic Scheduling** option at the bottom of the page as shown below.

![Scheduling tab](image)

2. In the **Scheduling** tab, select one or more days of the week to schedule the start of the synchronization.
3 Enter the time of day the synchronization occurs.
   In the above example the synchronization automatically begins every Saturday at 11:00 A.M.

4 Click Update to save the setting.

Running the Synchronization Process Now

To start the synchronization process on demand

After creating the External Reports Scheme Configuration policy, click Complete Synchronization under the Actions menu in the left pane.

IMPORTANT If an existing synchronization process has not completed, then ADM does not start the synchronization process.

Check the Status of the Synchronization Process

The ADM Console displays the status of the External Reports Schema Configuration policy that includes the following synchronization steps:

- Extraction of the database file.
- Creation of the database file.
- Posting the database files to the FTP site.

To check the status of the synchronization process

1 Log in to the ADM Console as admin user.

2 From the ADM Console, navigate to Connectors > Reports and click Status in the left pane.

3 The display pane lists the following parameters:
   - Last Sync. Time: The last time the external database was synchronized with the ADM repository. The time is based on the location of the appliance.
   - Synchronization Status: The status of the synchronization process.

NOTE Status of the synchronization process is one of the following:
   - Idle: The process is idle. Nothing is currently running.
   - Sync in progress: The ADM synchronization process is in progress.
   - Complete: The ADM synchronization is complete and the FTP files are loaded to the FTP Server.

   - Scheduled Synchronization Status: the status of the synchronization schedule.

NOTE Status of the synchronization schedule is one of the following:
   - Active: The synchronization schedule is set and runs at the scheduled time.
   - Not Active: There is no schedule associated with the policy.

   - Populated Elements: Lists the elements that were populated to the external database during the last synchronization.
   - Count: Gives the number of elements that were populated to the external database.

4 Click Log in the left pane to view the messages, describing the events of the synchronization process.
External Database Setup

To use the external database functionality you need to perform the following steps:

- “External Database System Requirements” on page 15
- “Installing and Configuring the External Database for Oracle” on page 16
- “Installing and Configuring the External Database for MS SQL” on page 20
- “Manually Loading Data to the External Database” on page 23
- “Checking the Status of the External Database Load Process” on page 23
- “Upgrading the External Database” on page 24

External Database System Requirements

To incorporate an external database into your ADM environment, you need the following components:

- Access to an FTP server.
- The external reporting database client has to be installed on the same server as the database.
- The hardware and software requirements as follows.

Minimum Hardware Requirements

Ensure that your system meets the following hardware requirements:

- 2 GB memory.
- Dual core 1.8 GHZ CPU.
- 500 MB for the client installation. If the data Zip files are retained on the server, more disk space is needed.
- 1 GB for the database instance.

Software Requirements

The software requirements for each type of database are as follows:

Oracle as the External Database

Ensure the system meets the following requirements when using Oracle as external database:

- Red Hat Enterprise Linux 4.0
- Oracle 10g R2 (10.2.0.1.0)
- Perl with the Archive::Zip module:
Perl is bundled with RHEL 4.0, but the Archive::Zip module for handling Zip files needs to be installed manually.

To verify that the Archive::Zip module is loaded, run the following:

```perl
perl -MArchive::Zip -e 'print'
```

If the result is: `Archive::Zip module is installed`, then the module is installed correctly.

If the result starts with: `can't find Archive/Zip`, then it is not installed.

To install this, execute the following command from the command prompt, with root permissions:

```bash
cpan -i Archive::Zip
```

- Perl with Expect module:
  
  To install, run the following:

  ```bash
  yum install perl-Expect
  ```

### MS SQL as the External Database

Ensure the system meets the following requirements when using MS SQL as the external database:

- Windows 2003 Service Pack 2
- MS SQL 2005
- psftp installed and added to the PATH env properties
- Perl with the Archive::Zip module

Common Perl distribution for Windows is ActivePerl from ActiveState. The necessary Archive::Zip module is bundled with ActivePerl.

### Installing and Configuring the External Database for Oracle

Use the following procedures to install and configure the external database for Oracle.

#### Install the Client-Side Script

**To install the client-side script**

1. Copy the ADM external database installation file (`adm_olap_v.X.X-XXX.zip`, where `v.X.X-XXX` is the ADM version and build number) from [http://downloads.vmware.com/Application Discovery Manager](http://downloads.vmware.com/Application Discovery Manager).
2. Log in to the Linux Server.
3. Change directories to the location of the installation file.
4. Unzip the `adm_olap_v.X.X-XXX.zip` to a directory of your choice.
   
   A directory named `adm_olap` is created with all the client-side scripts in it.
5. Change the `adm_olap` directory permission to allow read access to all users.
   
   To enable read permission, in Linux run:

   ```bash
   chmod 777 -R adm_olap
   ```
Create the External Reporting Database User

To create an external reporting database user

NOTE You must have database administrator permissions to create an adm_olap user and have $ORACLE_HOME/bin dir in the PATH to install and configure the host on which the external database is installed.

1 Switch to a user with oracle permissions for example:
   $ su – oracle
   $Password:
2 Change directory to unzip_path/adm_olap/schema/oracle/:
   cd unzip_path/adm_olap/schema/oracle/
   where unzip_path/adm_olap/schema/oracle/ is the directory in which the adm_olap_v.X.X-XXX.zip file was extracted.
3 Log in to the SQLplus server:
   $ sqlplus '/as sysdba'
   SQL*Plus: Release 10.1.0.2.0 - Production on Wed Dec 12 14:17:01 2007
   Copyright (c) 1982, 2004, Oracle. All rights reserved.
   Connected to:
   Oracle Database 10g Release 10.1.0.2.0 - Production
4 Verify that the instance is running, the database is mounted, and the listener is started.
5 Run the SQL script to create the adm_olap user:
   sql> @create_olap_user.sql
   User created.
   Grant succeeded.

NOTE The user created here must be the same user that was defined when setting up the External Reports Scheme Configuration policy. The External Reports Scheme Configuration defines the FTP server where the extracted data files are transferred and how often the synchronization occurs.

6 Type quit to exit the sqlplus prompt.

Create the adm_olap Tables

To create the adm_olap tables

NOTE The adm_olap schema creation is meant to be managed by the client Database Administrator (DBA). Simple SQL scripts are provided to support the basic ADM tables. To support additional information add external tables. Changes to the tables provided with ADM are overwritten with each load. The scripts are located where the adm_olap_v.X.X-XXX.zip file was extracted followed by/schema/oracle.

1 Log in to SQLplus as adm_olap:
   sqlplus adm_olap/adm_olap
2 Run the SQL script to create the tables:
   SQL> @create_tables.sql
3 Type quit to exit the sqlplus prompt.

NOTE Do not add constraints. With the constraints turned on, the loader fails to load the data.
Configure the Synchronization Process

To configure the synchronization process

1 Configure the sync.properties file located in the adm_olap directory:

   a Make a copy of the sync.properties.example file and name it sync.properties:

      ```
cp sync.properties.example sync.properties
      
      cp oracle.properties.example oracle.properties
      ```

   b In the sync.properties file, modify the FTP information. Enter the proper credentials, host remote directory, and whether or not to delete the files from the FTP server after the synchronization process completes:

      ```
      ftp.user=ftp-user
      ftp.password=ftp-password
      ftp.host=ftp-hostname
      ftp.dir=remote-dir
      ftp.local.dir=tmp
      # filter FTP files: consider only files starting with the prefix.
      ftp.file.prefix=adm_olap
      # delete files from FTP when done?
      ftp.delete.files=true
      ```

   c In the sync.properties file, ensure that the loader information points to the Oracle:

      ```
      loader.exec=./load_oracle.pl
      loader.dir=load
      loader.data.dir=data
      ```

   d In the sync.properties file, modify the archiving if necessary:

      ```
      # archive option will save a local copy of the compressed data at the specified directory
      archive=true
      archive.dir=archive
      
      #archive and archive.dir defines whether or not to store the zipped files locally for history tracking.
      
      # This can be used, for example, to remove indexes before the data loading
      # and re-create the indexes after the load is complete.
      #hook.before.load=
      #hook.after.load=
      ```

2 Configure the database loader for Oracle.

The Oracle loader script is configured using the oracle.properties file, that is located in the unzip_path/adm_olap/load/ directory. An example file, oracle.properties.example, has been provided in unzip_path/adm_olap/load/:

   a Make a copy of the oracle.properties.example file and name it oracle.properties:

      ```
cp oracle.properties.example oracle.properties
      ```

   b If you had defined a custom adm_olap user name and password when you had created the database user as described in “Create the External Reporting Database User” on page 17, then change the user name and password credentials accordingly:
user.id=adm_olap/adm_olap

c Change the delete.data.files parameter to =true if you want to delete the data files after they are loaded. If you keep the default, you need additional dataspace than defined in “External Database System Requirements” on page 15:

delete.data.files=false

3 Load the time data.

The current adm_olap schema has a time dimension table that is static. Fill the time table once and leave it unchanged thereafter. Perform the following steps before the loader first run:

cd adm_olap
unzip static_data/time.csv.zip -d data/

The time.csv data is loaded the next time the synchronization process runs.

4 Test the configuration to ensure that you have installed and configured the client correctly:

a Run a complete synchronization from the ADM Console as described in “Running the Synchronization Process Now” on page 13.

b After the upload completes, log in as a user with Oracle permissions, on the system on which the database is installed.

c Change directories to unzip_path/adm_olap/ and run the synchronization process:

cd <unzip_path>/adm_olap/
chmod 755 sync_adm_olap.pl
$ ./sync_adm_olap.pl

NOTE For secure FTP, use the sync_adm_olap_sftp_oracle.pl script instead of sync_adm_olap.pl

5 Schedule the synchronization process on Linux.

a Log in as a user with oracle permissions and enter:
crontab -e

b Verify the following information. It has to be part of the crontab for it to run properly:

PATH=... <oracle home full path/bin>
ORACLE_SID=>
ORACLE_HOME=>
<cron expression><install dir>/adm_olap/sync_adm_olap.pl >> <install dir>/adm_olap/log/sync.log 2>&1

NOTE For secure FTP, use the sync_adm_olap_sftp_oracle.pl script instead of sync_adm_olap.pl and for more information about cron expression please contact your unix admin.

The following is an example of a cron entry in Linux, where the process is scheduled to run after every 15 minutes:

PATH=...
ORACLE_SID=...
ORACLE_HOME=...
*/15 * * * * <install dir>/adm_olap/sync_adm_olap.pl >> <install dir>/adm_olap/log/sync.log 2>&1
Installing and Configuring the External Database for MS SQL

Use the following procedures to install and configure the external database for MS SQL.

Install the Client-Side Scripts

To install the client-side script

NOTE  You must have administrator privileges to install the client-side scripts on Windows.

1 Copy the ADM external database installation file (adm_olap_v.X.X-XXX.zip) from http://downloads.vmware.com/Application Discovery Manager.
2 Unzip the adm_olap_v.X.X-XXX.zip to a directory of your choice.

A directory named adm_olap is created with all the client-side scripts in it.

Install the External Database for MS SQL Server

To install the external database for MS SQL

NOTE  You can use either SQL Server Management Studio or SQLCMD command to run the installation scripts. The following steps demonstrate the commands using SQLCMD.

1 From the Windows Start menu, select Run and type cmd. The command prompt appears.
2 Change directories to unzip_path/adm_olap/schema/sqlserver/:
   cd <path>/adm_olap/schema/sqlserver
3 Run the create_database.sql script to create the ADM external database:
   sqlcmd -S <servername> -i create_database.sql
4 Run the create_olap_schema.sql script to create the adm_olap schema:
   sqlcmd -S <servername> -i create_olap_schema.sql
5 Run the create_adm_olap_login.sql script to create the login:
   sqlcmd -S <servername> -i create_adm_olap_login.sql
6 Run the create_user_for_login.sql script to create the login user:
   sqlcmd -S <servername> -i create_user_for_login.sql

NOTE  The user created here, must be the same user that was defined when setting up the External Reports Scheme Configuration policy. The External Reports Scheme Configuration policy defines the FTP server where the extracted data files are transferred and how often the synchronization occurs.

7 Run the create_tables.sql script to create the adm_olap tables:
   sqlcmd -S <servername> -i create_tables.sql

NOTE  Do not add constraints. With the constraints turned on, the loader fails to load the data.
Configure the Synchronization Process

To configure the synchronization process

1. Configure the sync.properties file located in the unzip_path/adm_olap directory.

   **NOTE** An example file, sync.properties.example, is provided in unzip_path/adm_olap to use as a template for configuring.

   a. Make a copy of the sync.properties.example file and name it sync.properties:

      ```
      copy sync.properties.example sync.properties
      ```

   b. In the sync.properties file, modify the FTP information. Enter the proper credentials, host remote directory, and whether or not to delete the files from the FTP server after the synchronization process completes:

      ```
      ftp.user=ftp-user
      ftp.password=ftp-password
      ftp.host=ftp-hostname
      ftp.dir=remote-dir
      ftp.local.dir=tmp
      # filter FTP files: consider only files starting with the prefix.
      ftp.file.prefix=adm_olap
      # delete files from FTP when done?
      ftp.delete.files=true
      ```

   **NOTE** Use the same user credentials when setting up the External Reports Scheme Configuration policy as described in “Creating the External Reports Scheme Configuration Policy” on page 11.

   c. In the sync.properties file, ensure that the loader information points to the MS SQL Server:

      ```
      loader.exec=load_sqlserver.pl
      loader.dir=load
      loader.data.dir=data
      ```

      where:

      loader.dir and loader.exec provide the location and script to run for loading the data.

      loader.data.dir is the data directory where the decompressed files are stored (relative to the unzip_path/adm_olap directory).

      unzip_path/adm_olap/data is the default directory and is left unchanged.

   d. In the sync.properties file, modify the archiving if necessary:

      ```
      # archive option will save a local copy of the compressed data at the specified directory
      archive=true
      archive.dir=archive
      ```

      archive and archive.dir defines whether or not to store the zipped files locally for history tracking.

      You can add the hooks if necessary. hook.before.load= and hook.after.load= allow custom behavior just before the data loading and after the loading is complete. Such hooks are used to add or drop indices, to add client specific data for loading together with ADM data, or any other desired actions. The property contains the path to an executable script:

      ```
      # This can be used, for example, to remove indexes before the data loading
      # and re-create the indexes after the load is complete.
      #hook.before.load=
      #hook.after.load=
      ```
2. Configure the database loader for MS SQL Servers.

   The SQL Server loader script is configured using the \sqlserver.properties file, that is located in the unzip_path/adm_olap/load/ directory. An example file, \sqlserver.properties.example, is provided in unzip_path/adm_olap/load/:

   a. Make a copy of the \sqlserver.properties.example file and name the file as \sqlserver.properties.

   b. Change the delete.data.files parameter to =true if you want to delete the data files after they are loaded. If you keep the default, you need additional dataspace than is defined in “External Database System Requirements” on page 15:

   ```
   delete.data.files=false
   ```

3. Load the time data.

   The current adm_olap schema has a time dimension table that is static. Fill the time table once and leave it unchanged thereafter. To have the loader load the time data upon initiation:

   Unzip the static_data/time.csv.zip file into the unzip_path/adm_olap/data directory.

   The time.csv data is loaded the next time the synchronization process runs.

4. Test your configuration:

   a. Run a complete synchronization from the ADM Console as described in “Running the Synchronization Process Now” on page 13.

   b. After the upload completes, log in as a user with administrator permissions, on the system on which the database is installed.

   c. From the Windows Start menu, select Run and type cmd. The command prompt appears.

   d. Change directories to unzip_path\adm_olap\ and run the synchronization process:

   ```
   cd <unzip_path>\adm_olap\\
   <perl_dir>\bin\perl.exe <install_dir>\adm_olap\sync_adm_olap.pl
   ```

   **NOTE** For secure FTP, use the sync_adm_olap_sftp_sqlserver.pl script instead of sync_adm_olap.pl

   To avoid id key verification for the first time by creating the id key, run psftp manually against the FTP server using command line psftp -pw password user@host. Type yes when prompted.

5. Schedule the synchronization process to run automatically for the MS SQL database:

   **NOTE** For the Windows scheduled task functionality, the flexibility to schedule tasks is limited. The schedules available are daily, weekly, monthly, and one time only, when the computer starts, or when the user logs in.

   a. From the Windows Start menu, navigate to Setting > Control Panel > Scheduled tasks.

   b. Double-click Add Scheduled Task to open the Schedule Task Wizard.

   c. Click Next to select the task you are scheduling.

   d. Browse to the unzip_path/adm_olap directory and select sync_adm_olap.pl.

   **NOTE** For secure FTP, use the sync_adm_olap_sftp_sqlserver.pl script instead of sync_adm_olap.pl

   e. Enter a name for the sync_adm_olap.pl task schedule.

   f. Select how often you want to run the task from the options and click Next.

   g. Depending on the interval selected, define the time, days, and the date to start the task.
Enter the user credentials of the host administrator to ensure that environment settings are configured correctly.

If the user name and password is recognized, a message is displayed confirming the task has been scheduled.

Select **Open advanced properties for this task when I click Finish** option.

Click **Finish**.

From the Windows **Start** menu, navigate to **Setting > Control Panel > Scheduled tasks**.

Right-click the task and click **Properties**. The task Properties dialog appears.

The script is a Perl script and has to be run by Perl. Change the **Run** field to obtain the full path of the perl executable. For example:

```
C:\Perl\bin\perl.exe <install_dir>\adm_olap\sync_adm_olap.pl
```

where: `C:\Perl\bin\` is the Perl installation directory.

If Perl is installed in another location, be sure to use the correct path.

View the loader logs under `unzip_path\adm_olap\load\sqlserver\log`

---

**Manually Loading Data to the External Database**

The data files are manually loaded to the external database.

**To load the zip files manually for Oracle**

1. Unzip the contents of the zip file to the `unzip_path/adm_olap/data` directory:
   
   ```
   unzip adm_olap_v.X.X-XXX.zip -d adm_olap/data/
   ```

2. Change directories to the loader base directory and run the loader script:
   
   ```
   cd <unzip_path>/adm_olap/load/
   ./load_oracle.pl
   ```

**To load the zip files manually for MS SQL Server**

1. From Windows Explorer, copy the `adm_olap_v.X.X-XXX.zip` file to `unzip_path\adm_olap\data` folder.

2. Right-click the `adm_olap_v.X.X-XXX.zip` file and unzip it to the same location.

3. From the Windows **Start** menu, select **Run** and type `cmd`. The command prompt appears.

4. Change directories to the loader base directory and run the loader script:
   
   ```
   cd <unzip_path>\adm_olap\load\load_sqlserver.pl
   ```

**Checking the Status of the External Database Load Process**

Use the following log files to check the status of the load process to the external database:

- For Oracle, check: `/unzip_path/adm_olap/load/oracle/log`
- For MS.SQL, check: `unzip_path\adm_olap\load\sqlserver\log`
Upgrading the External Database

**IMPORTANT** The upgrade process requires a complete uninstall of the existing database version and an installation of the new database version.

Perform the following steps to upgrade the external database.

### Upgrading for MS SQL

When upgrading the external database for MS SQL:

1. Optionally, record the **Scheduled Task** configuration:
   a. From the Windows Start menu, navigate to Setting > Control Panel > Scheduled tasks and open the sync_adm_olap task.
   b. Record the configuration for later use.

2. Delete the sync_adm_olap task.
   a. From the Windows Start menu, navigate to Setting > Control Panel > Scheduled tasks and open the sync_adm_olap task.
   b. Right-click the sync_adm_olap task and click Delete.

3. Ensure that synchronization is not currently running as follows:
   a. Log in to the ADM Console as admin user.
   b. From the ADM Console, navigate to Connectors > Reports > External Reports Scheme Integration Status.
   c. If Synchronization Status indicates synchronization in progress, wait till it completes and then proceed with the upgrade.

**IMPORTANT** The following step clears the data in the external report database. The data is recovered once you perform a full synchronization after the upgrade is complete.

Ensure that SQL scripts are associated with SQL Server Management Studio before continuing.

4. Execute the following SQL scripts located in the adm_olap/scheme/sqlserver folder using the user/password adm_olap/adm_olap (To execute, double-click the script):
   - drop_constraints.sql
   - truncate_data.sql
   - drop_tables.sql

5. Execute the following SQL scripts located in the adm_olap/scheme/sqlserver folder using a user with administrator permissions (To execute, double-click the script):
   - drop_olap_schema.sql
   - drop_database.sql

6. Backup the old adm_olap folder that includes the sync.properties and sqlserver.properties files (It contains your configuration settings).

7. Proceed with the installation instructions in “Installing and Configuring the External Database for MS SQL” on page 20.

**CAUTION** Do not copy the new configuration files over the backed up configuration files. Use the backed up configuration files only as a reference to manually update the new file.
Upgrading for Oracle

When upgrading the external database for Oracle:

1. Ensure the `sync_adm_olap` task is removed from the cron job as follows:
   a. At the command prompt, type:
      ```
crontab -e
```
   b. Delete the task from the file.
   c. Save and exit.

2. Log in to the ADM Console as admin user.

3. From the ADM Console, navigate to Connectors > Reports > External Reports Scheme Integration Status and check if the **Synchronization Status** is complete. If the status is **sync in progress**, wait for it to complete.

4. Backup the folder in which you originally installed the `adm_olap` files. This folder includes the `sync.properties` and the `oracle.properties` files, that contains your configuration settings.

   **IMPORTANT** The data in the external report database is lost as a result of the following step. The data is recovered once you perform a full synchronization after the upgrade is complete.

5. Execute the following scripts located in `adm_olap/scheme/oracle`:
   - `sqlplus adm_olap/adm_olap @ drop_constraints.sql`
   - `sqlplus adm_olap/adm_olap @ truncate_data.sql`
   - `sqlplus adm_olap/adm_olap @ drop_tables.sql`

6. Backup the old `adm_olap` folder (includes the `sync.properties` and `oracle.properties` files, and contains your configuration settings).

7. Proceed with the installation instructions in “Installing and Configuring the External Database for Oracle” on page 16.

   **CAUTION** Do not copy the new configuration files over the backed up configuration files. Use the backed up configuration files only as a reference to manually update the new file.
This chapter illustrates the table structure and dependencies within the ADM external database. It also describes the tables within the external database in alphabetical order:

This chapter includes the following topics:

- “Overview of External Database Tables” on page 28
- “Address” on page 29
- “Change_Log” on page 29
- “CPU” on page 29
- “Database” on page 30
- “Dependency” on page 30
- “Group_Membership” on page 31
- “Installed_Software” on page 32
- “IP_Route” on page 32
- “Java_EE” on page 33
- “Network_Element” on page 33
- “Network_Interface” on page 34
- “Port” on page 35
- “Property” on page 35
- “Service” on page 36
- “Storage_Device” on page 36
- “Switch_Address_Forwarding” on page 37
- “Time” on page 37
- “Usage_Metric” on page 38
- “Usage_Metric_Baseline” on page 38
- “Virtual_System_Settings” on page 38
Overview of External Database Tables

This chapter provides information about the external database tables.

Database Structure and Dependencies

Figure 4-1 illustrates the structure and dependencies of the ADM external database. The arrows originate from a foreign key and arrow head points to the table keys.

**Figure 4-1. External Database Structure and Dependencies**

![ERD diagram showing relationships between tables]

**NOTE** Some of the commonly accessed data is duplicated in many tables to avoid redundant joins.

You can join the relation ENTITY_ID (in Group_Membership table, Usage_Metric table, and so on) with Network_Element table, Service_table, and Dependency table although it is not shown in the ERD as a foreign key.

For display purpose, some data is aggregated to a comma-separated list. For example:

- **HOSTNAMES:** A flat list of one or more host names.
- **HOST_IPS:** A flat list of all IPs of a host.
- **GROUPS:** A flat list of groups the entity belongs to.
Address

Address table contains information about a network element consisting of the host name, IP address, and MAC address. Table 4-1 lists the contents of the Address table.

Table 4-1. Address Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Address table (primary key)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>NIC_ID</td>
<td>NUMBER(19)</td>
<td>Network interface identifier (foreign key)</td>
</tr>
<tr>
<td>MAC</td>
<td>VARCHAR2(100)</td>
<td>MAC address</td>
</tr>
<tr>
<td>IP</td>
<td>VARCHAR2(100)</td>
<td>IP address</td>
</tr>
<tr>
<td>HOSTNAME</td>
<td>VARCHAR2(1024)</td>
<td>Host name</td>
</tr>
</tbody>
</table>

Change_Log

Change_Log table captures the changes to entity data. Table 4-2 lists the contents of the Change_Log table.

Table 4-2. Change_Log Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Change_Log table (primary key)</td>
</tr>
<tr>
<td>ENTITY_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>CHANGE_TIME</td>
<td>TIMESTAMP(6)</td>
<td>Time of change</td>
</tr>
<tr>
<td>CHANGE_TYPE</td>
<td>VARCHAR2(10)</td>
<td>Type of change (Added, Modified, Removed)</td>
</tr>
<tr>
<td>PATH</td>
<td>VARCHAR(22048)</td>
<td>Path to the changed data</td>
</tr>
<tr>
<td>DISPLAY_NAME</td>
<td>VARCHAR2(255)</td>
<td>Name of the data property changed</td>
</tr>
<tr>
<td>OLD_VALUE</td>
<td>VARCHAR2(1024)</td>
<td>Old value</td>
</tr>
<tr>
<td>NEW_VALUE</td>
<td>VARCHAR2(1024)</td>
<td>New value</td>
</tr>
</tbody>
</table>

CPU

CPU table contains the processor information for a network element. The information is identical to the information in the network element table unless the machine has multiple types of processors. Table 4-3 lists the contents of the CPU table.

Table 4-3. CPU Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the CPU table (primary key)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>MODEL</td>
<td>VARCHAR2(256)</td>
<td>Processor model</td>
</tr>
<tr>
<td>VENDOR</td>
<td>VARCHAR2(256)</td>
<td>Processor vendor</td>
</tr>
<tr>
<td>FAMILY</td>
<td>VARCHAR2(256)</td>
<td>Processor family</td>
</tr>
<tr>
<td>CLOCK_SPEED_HZ</td>
<td>NUMBER(19)</td>
<td>Max clock speed in Hertz</td>
</tr>
<tr>
<td>CACHE_SIZE</td>
<td>NUMBER(19)</td>
<td>Cache size in Bytes</td>
</tr>
<tr>
<td>STEPPING</td>
<td>VARCHAR2(256)</td>
<td>Processor version</td>
</tr>
</tbody>
</table>
**Database**

Database table contains database server and instance information. Several instances might exist for each server. **Table 4-4** lists the contents of the Database table.

**Table 4-4. Database Table**

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Database table (primary key)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>SERVICE_ID</td>
<td>NUMBER(19)</td>
<td>Service identifier (foreign key)</td>
</tr>
<tr>
<td>VENDOR</td>
<td>VARCHAR2(1024)</td>
<td>Vendor of the database</td>
</tr>
<tr>
<td>SERVER_NAME</td>
<td>VARCHAR2(1024)</td>
<td>Name of the database server</td>
</tr>
<tr>
<td>INSTANCE</td>
<td>VARCHAR2(256)</td>
<td>Database instance</td>
</tr>
<tr>
<td>IS_AUTO_START</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the database as auto start (values: Y or N)</td>
</tr>
</tbody>
</table>

**Dependency**

Dependency table represents the relationship between a source host or service and target host or service. **Table 4-5** lists the contents of the Dependency table.

**Table 4-5. Dependency Table**

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Dependency table (primary key)</td>
</tr>
<tr>
<td>DEPENDENCY_TYPE</td>
<td>VARCHAR2(100)</td>
<td>Network connection or documented dependency</td>
</tr>
<tr>
<td>PROTOCOL</td>
<td>VARCHAR2(1024)</td>
<td>Network connection protocol (for example, HTTP, FTP, and so on)</td>
</tr>
<tr>
<td>PROTOCOL_VERSION</td>
<td>VARCHAR2(1024)</td>
<td>Version of the protocol</td>
</tr>
<tr>
<td>TRANSPORT_PROTOCOL</td>
<td>VARCHAR2(100)</td>
<td>Transport protocol (values: TCP or UDP)</td>
</tr>
<tr>
<td>URL</td>
<td>VARCHAR2(512)</td>
<td>Documented dependency URL</td>
</tr>
<tr>
<td>DOC_DEP_TYPE</td>
<td>VARCHAR2(256)</td>
<td>Type of the documented dependency</td>
</tr>
<tr>
<td>SRC_IP</td>
<td>VARCHAR2(100)</td>
<td>Source IP address of the dependency</td>
</tr>
<tr>
<td>SRC_PORT</td>
<td>NUMBER(19)</td>
<td>Source port of the dependency</td>
</tr>
<tr>
<td>SRC_SERVICE_NAME</td>
<td>VARCHAR2(1024)</td>
<td>Source service name</td>
</tr>
<tr>
<td>SRC_PRIMARY_HOSTNAME</td>
<td>VARCHAR2(1024)</td>
<td>Source primary host name (see Network Element table)</td>
</tr>
<tr>
<td>SRC_PRIMARY_HOST_IP</td>
<td>VARCHAR2(100)</td>
<td>Source primary IP address (see Network Element table)</td>
</tr>
<tr>
<td>SRC_HOSTNAMES</td>
<td>VARCHAR2(4000)</td>
<td>Source host names (see Network Element table)</td>
</tr>
<tr>
<td>SRC_HOST_IPS</td>
<td>VARCHAR2(4000)</td>
<td>Source IP addresses (see Network Element table)</td>
</tr>
</tbody>
</table>
Table 4-5. Dependency Table (Continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRC_NE_ID</td>
<td>NUMBER(19)</td>
<td>Source network element identifier (foreign key)</td>
</tr>
<tr>
<td>SRC_SERVICE_ID</td>
<td>NUMBER(19)</td>
<td>Source service identifier (foreign key)</td>
</tr>
<tr>
<td>SRC_MODULE_ID</td>
<td>NUMBER(19)</td>
<td>Source module identifier (java_ee, database), usually of a documented dependency (foreign key)</td>
</tr>
<tr>
<td>TRG_IP</td>
<td>VARCHAR2(100)</td>
<td>Target IP address of the dependency</td>
</tr>
<tr>
<td>TRG_PORT</td>
<td>NUMBER(19)</td>
<td>Target port of the dependency</td>
</tr>
<tr>
<td>TRG_SERVICE_NAME</td>
<td>VARCHAR2(1024)</td>
<td>Target service name</td>
</tr>
<tr>
<td>TRG_PRIMARY_HOSTNAME</td>
<td>VARCHAR2(1024)</td>
<td>Target primary host name (see Network Element table)</td>
</tr>
<tr>
<td>TRG_PRIMARY_HOST_IP</td>
<td>VARCHAR2(100)</td>
<td>Target primary IP address (see Network Element table)</td>
</tr>
<tr>
<td>TRG_HOSTNAMES</td>
<td>VARCHAR2(4000)</td>
<td>Target host names (see Network Element table)</td>
</tr>
<tr>
<td>TRG_HOST_IPS</td>
<td>VARCHAR2(4000)</td>
<td>Target IP addresses (see Network Element table)</td>
</tr>
<tr>
<td>TRG_NE_ID</td>
<td>NUMBER(19)</td>
<td>Target network element identifier</td>
</tr>
<tr>
<td>TRG_SERVICE_ID</td>
<td>NUMBER(19)</td>
<td>Target service identifier</td>
</tr>
<tr>
<td>TRG_MODULE_ID</td>
<td>NUMBER(19)</td>
<td>Target module (java_ee, database), usually of a documented dependency</td>
</tr>
<tr>
<td>GROUPS</td>
<td>VARCHAR2(4000)</td>
<td>Comma-separated list of all groups the element belongs to</td>
</tr>
<tr>
<td>DISCOVERY_TIME</td>
<td>TIMESTAMP(6)</td>
<td>Time the element was first added to the system</td>
</tr>
<tr>
<td>KNOWLEDGE_LEVEL</td>
<td>NUMBER(10)</td>
<td>Indicates the level of knowledge of the dependency. Value of zero indicates speculative information only (used to filter out speculative information)</td>
</tr>
</tbody>
</table>

**Group_Membership**

The groups within ADM contain entities as members. Group_Membership table identifies an entity (network element, service, dependency) as a member of a group. Table 4-6 lists the contents of the Group_Membership table.

Table 4-6. Group_Membership Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Group identifier is unique to groups but not to entity memberships (primary key)</td>
</tr>
<tr>
<td>GROUP_TYPE</td>
<td>VARCHAR2(255)</td>
<td>Group type</td>
</tr>
<tr>
<td>GROUP_NAME</td>
<td>VARCHAR2(255)</td>
<td>Group name</td>
</tr>
<tr>
<td>GROUP_DESCRIPTION</td>
<td>VARCHAR2(1024)</td>
<td>Group description</td>
</tr>
<tr>
<td>GROUP_CREATION_TIME</td>
<td>TIMESTAMP(6)</td>
<td>Group creation time</td>
</tr>
<tr>
<td>GROUP_CREATOR</td>
<td>VARCHAR2(255)</td>
<td>Group creator user name</td>
</tr>
<tr>
<td>GROUP_MODIFICATION_TIME</td>
<td>TIMESTAMP(6)</td>
<td>Group last modification time</td>
</tr>
<tr>
<td>GROUP_MODIFIER</td>
<td>VARCHAR2(255)</td>
<td>Group last modifier user name</td>
</tr>
<tr>
<td>ENTITY_ID</td>
<td>NUMBER(19)</td>
<td>The data contains foreign key to entity types: Network Element, Service, Dependency</td>
</tr>
</tbody>
</table>
**Installed_Software**

Installed_Software table contains the information of software installed on a network element. Installed software can appear as both installed software and as a service on the network. Software information might also exist in the service table. Table 4-7 lists the contents of the Installed_Software table.

### Table 4-7. Installed_Software Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Installed_Software table (primary key)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>NAME</td>
<td>VARCHAR2(256)</td>
<td>Software name</td>
</tr>
<tr>
<td>VERSION</td>
<td>VARCHAR2(256)</td>
<td>Software version</td>
</tr>
<tr>
<td>RELEASE</td>
<td>VARCHAR2(256)</td>
<td>Software release number</td>
</tr>
<tr>
<td>VENDOR</td>
<td>VARCHAR2(256)</td>
<td>Software vendor</td>
</tr>
<tr>
<td>INSTALL_PATH</td>
<td>VARCHAR2(1024)</td>
<td>Path of the installation</td>
</tr>
<tr>
<td>SHARED_INSTALL_PATH</td>
<td>VARCHAR2(1024)</td>
<td>Path of the shared installation</td>
</tr>
<tr>
<td>INSTALL_DATE</td>
<td>TIMESTAMP(6)</td>
<td>Date of installation</td>
</tr>
<tr>
<td>TYPE</td>
<td>VARCHAR2(256)</td>
<td>Type of software (for example, application, driver, patch and so on)</td>
</tr>
</tbody>
</table>

**IP_Route**

IP_Route table contains IP (layer 3) routing information for a network element. Table 4-8 lists the contents of the IP_Route table.

### Table 4-8. IP_Route Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the IP_Route table (primary key)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>NIC_ID</td>
<td>NUMBER(19)</td>
<td>Network interface identifier (foreign key)</td>
</tr>
<tr>
<td>DEST_IP</td>
<td>VARCHAR2(100)</td>
<td>Destination IP address</td>
</tr>
<tr>
<td>DEST_MASK</td>
<td>VARCHAR2(100)</td>
<td>Destination subnet mask</td>
</tr>
<tr>
<td>GATEWAY</td>
<td>VARCHAR2(100)</td>
<td>Gateway IP address</td>
</tr>
<tr>
<td>METRIC</td>
<td>NUMBER(10)</td>
<td>Priority value</td>
</tr>
<tr>
<td>IS_LOCAL</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the IP address as local (values: Y or N)</td>
</tr>
<tr>
<td>GATEWAY_NE_ID</td>
<td>NUMBER(19)</td>
<td>Gateway network element identifier (foreign key)</td>
</tr>
</tbody>
</table>
Java_EE

Java_EE table contains Java Enterprise Edition components such as Application Server, JEE Application, and modules (jar, war, ear, and so on). Table 4-9 lists the contents of the Java_EE table.

Table 4-9. Java_EE Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Java_EE table (primary key)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>SERVICE_ID</td>
<td>NUMBER(19)</td>
<td>Service identifier (foreign key)</td>
</tr>
<tr>
<td>SERVER_NAME</td>
<td>VARCHAR2(256)</td>
<td>Application server name</td>
</tr>
<tr>
<td>SERVER_CONFIG_PATH</td>
<td>VARCHAR2(512)</td>
<td>Server configuration path</td>
</tr>
<tr>
<td>APP_NAME</td>
<td>VARCHAR2(256)</td>
<td>JEE application name</td>
</tr>
<tr>
<td>APP_DESCRIPTION</td>
<td>VARCHAR2(1024)</td>
<td>JEE application description</td>
</tr>
<tr>
<td>APP_PATH</td>
<td>VARCHAR2(1024)</td>
<td>JEE application path</td>
</tr>
<tr>
<td>MODULE_NAME</td>
<td>VARCHAR2(256)</td>
<td>JEE module name</td>
</tr>
<tr>
<td>MODULE_TYPE</td>
<td>VARCHAR2(256)</td>
<td>JEE module type (for example, jar, war, ear and so on)</td>
</tr>
<tr>
<td>MODULE_DEPLOYMENT_DESC</td>
<td>VARCHAR2(256)</td>
<td>Deployment descriptor</td>
</tr>
<tr>
<td>MODULE_CONTEXT_ROOT</td>
<td>VARCHAR2(256)</td>
<td>Context root (war only)</td>
</tr>
</tbody>
</table>

Network_Element

Network_Element table contains computer system information, which could be any combination of host, router, and switch. Table 4-10 lists the contents of the Network_Element table.

Table 4-10. Network_Element Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Network_Element table (primary key)</td>
</tr>
<tr>
<td>PRIMARY_HOSTNAME</td>
<td>VARCHAR2(1024)</td>
<td>Primary host name</td>
</tr>
<tr>
<td>PRIMARY_HOST_IP</td>
<td>VARCHAR2(100)</td>
<td>Primary IP address of the host</td>
</tr>
<tr>
<td>HOSTNAMES</td>
<td>VARCHAR2(4000)</td>
<td>Comma-separated list of all host names</td>
</tr>
<tr>
<td>HOST_IPS</td>
<td>VARCHAR2(4000)</td>
<td>Comma-separated list of all IP addresses</td>
</tr>
<tr>
<td>GROUPS</td>
<td>VARCHAR2(4000)</td>
<td>Comma-separated list of all groups the network element belongs to</td>
</tr>
<tr>
<td>DISCOVERY_TIME</td>
<td>TIMESTAMP(6)</td>
<td>Time the network element was first added to the system</td>
</tr>
<tr>
<td>IS_HOST</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the network element as a host (values: Y or N)</td>
</tr>
<tr>
<td>IS_ROUTER</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the network element as a router (values: Y or N)</td>
</tr>
<tr>
<td>IS_SWITCH</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the network element as a switch (values: Y or N)</td>
</tr>
<tr>
<td>IS_VIRT_CONTAINER</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the network element as virtual container (values: Y or N)</td>
</tr>
<tr>
<td>IS_VIRTUAL</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the network element as a virtual machine (values: Y or N)</td>
</tr>
<tr>
<td>HW_SERIAL</td>
<td>VARCHAR2(256)</td>
<td>Serial number of the target host</td>
</tr>
</tbody>
</table>
### Network_Interface Table

Network_Interface table contains network interface card or switch port information. Table 4-11 lists the contents of the Network_Interface table.

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Network_Interface table (primary key)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>IS_LOCAL</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the IP address assigned to the NIC as local (values: Y or N)</td>
</tr>
<tr>
<td>MAC_ADDRESS</td>
<td>VARCHAR2(256)</td>
<td>MAC address of the NIC</td>
</tr>
</tbody>
</table>

---

**Table 4-10. Network_Element Table (Continued)**

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HW_UUID</td>
<td>VARCHAR2(256)</td>
<td>UUID of the target host</td>
</tr>
<tr>
<td>HW_VENDOR</td>
<td>VARCHAR2(256)</td>
<td>Hardware vendor</td>
</tr>
<tr>
<td>HW_MODEL</td>
<td>VARCHAR2(256)</td>
<td>Hardware model</td>
</tr>
<tr>
<td>HW_DESCRIPTION</td>
<td>VARCHAR2(1024)</td>
<td>Description of the hardware</td>
</tr>
<tr>
<td>OS_NAME</td>
<td>VARCHAR2(256)</td>
<td>Operating system name</td>
</tr>
<tr>
<td>OS_FAMILY</td>
<td>VARCHAR2(256)</td>
<td>Operating system family</td>
</tr>
<tr>
<td>OS_TYPE</td>
<td>VARCHAR2(256)</td>
<td>Operating system type</td>
</tr>
<tr>
<td>OS_VERSION</td>
<td>VARCHAR2(256)</td>
<td>Operating system version</td>
</tr>
<tr>
<td>OS_VENDOR</td>
<td>VARCHAR2(256)</td>
<td>Operating system vendor</td>
</tr>
<tr>
<td>OS_BUILD</td>
<td>VARCHAR2(256)</td>
<td>Operating system build</td>
</tr>
<tr>
<td>OS_BUILD_TYPE</td>
<td>VARCHAR2(256)</td>
<td>Operating system build type</td>
</tr>
<tr>
<td>OS_SERVICE_PACK</td>
<td>VARCHAR2(256)</td>
<td>Operating system service pack</td>
</tr>
<tr>
<td>OS_LANGUAGE</td>
<td>VARCHAR2(256)</td>
<td>Operating system language</td>
</tr>
<tr>
<td>OS_KERNEL_VERSION</td>
<td>VARCHAR2(256)</td>
<td>Operating system kernel version</td>
</tr>
<tr>
<td>OS_KERNEL_RELEASE</td>
<td>VARCHAR2(256)</td>
<td>Operating system kernel release</td>
</tr>
<tr>
<td>OS_SWAP_MEMORY</td>
<td>NUMBER(19)</td>
<td>Swap memory size used by the operating system in Bytes</td>
</tr>
<tr>
<td>OS_PHYS_MEMORY</td>
<td>NUMBER(19)</td>
<td>Physical memory as detected by the operating system in Bytes</td>
</tr>
<tr>
<td>OS_REGISTRATION_KEY</td>
<td>VARCHAR2(256)</td>
<td>Operating system registration key</td>
</tr>
<tr>
<td>OS_SERIAL_NUMBER</td>
<td>VARCHAR2(256)</td>
<td>Operating system serial number</td>
</tr>
<tr>
<td>CPU_MODEL</td>
<td>VARCHAR2(256)</td>
<td>Processor model</td>
</tr>
<tr>
<td>CPU_VENDOR</td>
<td>VARCHAR2(256)</td>
<td>Processor vendor</td>
</tr>
<tr>
<td>CPU_FAMILY</td>
<td>VARCHAR2(256)</td>
<td>Processor family</td>
</tr>
<tr>
<td>CPU_CLOCK_SPEED_HZ</td>
<td>NUMBER(19)</td>
<td>Maximum processor speed in Hertz</td>
</tr>
<tr>
<td>CPU_CACHE_SIZE</td>
<td>NUMBER(19)</td>
<td>Processor cache size in Bytes</td>
</tr>
<tr>
<td>CPU_STEPPING</td>
<td>VARCHAR2(256)</td>
<td>Processor version</td>
</tr>
<tr>
<td>CPU_DESCRIPTION</td>
<td>VARCHAR2(256)</td>
<td>Processor description</td>
</tr>
<tr>
<td>CPU_COUNT</td>
<td>NUMBER(10)</td>
<td>Processor count as detected by the operating system (multi-core CPU is usually detected by the operating system as several separate CPUs)</td>
</tr>
</tbody>
</table>
**Table 4-11. Network_Interface Table (Continued)**

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODEL</td>
<td>VARCHAR2(256)</td>
<td>Model of the NIC</td>
</tr>
<tr>
<td>VENDOR</td>
<td>VARCHAR2(256)</td>
<td>Vendor of the NIC</td>
</tr>
<tr>
<td>TECHNOLOGY</td>
<td>VARCHAR2(256)</td>
<td>Networking technology used (for example, WAN, Ethernet and so on)</td>
</tr>
<tr>
<td>SPEED_BIT_PS</td>
<td>VARCHAR2(256)</td>
<td>Speed in bits per second</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td>VARCHAR2(256)</td>
<td>Description of the NIC</td>
</tr>
<tr>
<td>OP_STATUS</td>
<td>VARCHAR2(256)</td>
<td>Operation status</td>
</tr>
<tr>
<td>IS_VIRTUAL</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the NIC as virtual (values: Y or N)</td>
</tr>
<tr>
<td>IS_TRUNK</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying VLAN trunking (values: Y or N)</td>
</tr>
<tr>
<td>IS_DEFAULT</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the NIC as default (values: Y or N)</td>
</tr>
<tr>
<td>PORT_LABEL</td>
<td>VARCHAR2(256)</td>
<td>Switch port label</td>
</tr>
<tr>
<td>MODULE_LABEL</td>
<td>VARCHAR2(256)</td>
<td>Switch module label</td>
</tr>
<tr>
<td>V_LAN</td>
<td>NUMBER(10)</td>
<td>Number of Virtual LANs</td>
</tr>
</tbody>
</table>

SPEED_BIT_PS was speed_mb in version 5.3 of ADM

**Port**

Port table represents a logical network port to which a service binds to (for example, an HTTP server binds to port 80). Table 4-12 lists the contents of the Port table.

**Table 4-12. Port Table**

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Port table (primary key)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>SERVICE_ID</td>
<td>NUMBER(19)</td>
<td>Service identifier (foreign key)</td>
</tr>
<tr>
<td>IP</td>
<td>VARCHAR2(100)</td>
<td>IP address</td>
</tr>
<tr>
<td>PORT</td>
<td>NUMBER(5)</td>
<td>Port number</td>
</tr>
<tr>
<td>TRANSPORT_PROTOCOL</td>
<td>VARCHAR2(100)</td>
<td>Transport protocol (values: TCP or UDP)</td>
</tr>
</tbody>
</table>

**Property**

Property table contains general purpose properties (name-value pairs), such as URLs on HTTP servers and tables on database-related dependencies. Table 4-13 lists the contents of the Property table.

**Table 4-13. Property Table**

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Property table (primary key)</td>
</tr>
<tr>
<td>ENTITY_ID</td>
<td>NUMBER(19)</td>
<td>The data contains foreign key to entity types: Network Element, Service, Dependency</td>
</tr>
<tr>
<td>NAME</td>
<td>VARCHAR2(256)</td>
<td>Property name</td>
</tr>
<tr>
<td>VALUE</td>
<td>VARCHAR2(1028)</td>
<td>Property value</td>
</tr>
</tbody>
</table>
Service

Service table contains information about a software service running on the network element (usually exposed through a network port). Table 4-14 lists the contents of the Service table.

Table 4-14. Service Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Service table (primary key)</td>
</tr>
<tr>
<td>NAME</td>
<td>VARCHAR2(1024)</td>
<td>Service name if known (null if unknown)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>PRIMARY_HOSTNAME</td>
<td>VARCHAR2(1024)</td>
<td>Primary host name of the containing host (see Network Element table)</td>
</tr>
<tr>
<td>PRIMARY_HOST_IP</td>
<td>VARCHAR2(100)</td>
<td>Primary IP address of the containing host (see Network Element table)</td>
</tr>
<tr>
<td>HOSTNAMES</td>
<td>VARCHAR2(4000)</td>
<td>Comma-separated list of all hosts (see Network Element table)</td>
</tr>
<tr>
<td>HOST_IPS</td>
<td>VARCHAR2(4000)</td>
<td>Comma-separated list of all host IP addresses (see Network Element table)</td>
</tr>
<tr>
<td>PORTS</td>
<td>VARCHAR2(4000)</td>
<td>Ports that consume the service</td>
</tr>
<tr>
<td>GROUPS</td>
<td>VARCHAR2(4000)</td>
<td>Comma-separated list of all groups the element belongs to</td>
</tr>
<tr>
<td>DISCOVERY_TIME</td>
<td>TIMESTAMP(6)</td>
<td>Time the element was first added to the system</td>
</tr>
<tr>
<td>VENDOR</td>
<td>VARCHAR2(1024)</td>
<td>Service vendor</td>
</tr>
<tr>
<td>VERSION</td>
<td>VARCHAR2(1024)</td>
<td>Service version</td>
</tr>
<tr>
<td>CATEGORY</td>
<td>VARCHAR2(100)</td>
<td>Category of the service (for example, application server, database server, web server, and so on)</td>
</tr>
<tr>
<td>TIER</td>
<td>VARCHAR2(100)</td>
<td>Tier of the service (for example, application, database, and so on)</td>
</tr>
<tr>
<td>INSTALL_PATH</td>
<td>VARCHAR2(512)</td>
<td>Installation path</td>
</tr>
<tr>
<td>SHARED_INSTALL_PATH</td>
<td>VARCHAR2(512)</td>
<td>Shared installation path</td>
</tr>
<tr>
<td>IS_UNIDENTIFIED</td>
<td>VARCHAR2(1)</td>
<td>Flag indicating the service as unidentified service (values: Y or N)</td>
</tr>
<tr>
<td>IS_CLIENT</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the service as client service (values: Y or N)</td>
</tr>
<tr>
<td>KNOWLEDGE_LEVEL</td>
<td>NUMBER(10)</td>
<td>Indicates the level of knowledge of the service. A value of zero indicates speculative information only (used to filter out speculative information)</td>
</tr>
</tbody>
</table>

Storage_Device

Storage_Device table represents network element storage device. It contains both logical drives (such as partitions) and physical drives. Table 4-15 lists the contents of the Storage_Device table.

Table 4-15. Storage_Device Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Storage_Device table (primary key)</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>IS_LOGICAL</td>
<td>VARCHAR2(1)</td>
<td>Flag identifying the volume as logical volume (values: Y or N)</td>
</tr>
</tbody>
</table>
Switch_Address_Forwarding

Switch_Address_Forwarding table contains address forwarding information of a switch (layer 2). Table 4-16 lists the contents of the Switch_Address_Forwarding table.

Table 4-16. Switch_Address_Forwarding Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Switch_Address_Forwarding table</td>
</tr>
<tr>
<td>NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier (foreign key)</td>
</tr>
<tr>
<td>NIC_ID</td>
<td>NUMBER(19)</td>
<td>Network interface identifier (foreign key)</td>
</tr>
<tr>
<td>MODULE</td>
<td>VARCHAR2(256)</td>
<td>Switch module</td>
</tr>
<tr>
<td>PORT</td>
<td>VARCHAR2(256)</td>
<td>Switch port</td>
</tr>
<tr>
<td>MAC</td>
<td>VARCHAR2(100)</td>
<td>Forward MAC address</td>
</tr>
<tr>
<td>V_LAN</td>
<td>NUMBER(10)</td>
<td>Virtual LAN</td>
</tr>
<tr>
<td>TRG_NE_ID</td>
<td>NUMBER(19)</td>
<td>Forward network element identifier (foreign key)</td>
</tr>
</tbody>
</table>

Time

Time table contains the time information. Table 4-17 lists the contents of the Time table.

Table 4-17. Time Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIME_ID</td>
<td>NUMBER(19)</td>
<td>Human readable number: YYYYMMDDHH (primary key)</td>
</tr>
<tr>
<td>YEAR</td>
<td>NUMBER(10)</td>
<td>Year (YYYY)</td>
</tr>
<tr>
<td>QUARTER</td>
<td>NUMBER(10)</td>
<td>Quarter (1-4)</td>
</tr>
<tr>
<td>QUARTER_NAME</td>
<td>VARCHAR2(50)</td>
<td>Quarter name (Q1-2008)</td>
</tr>
<tr>
<td>MONTH</td>
<td>NUMBER(10)</td>
<td>Month number (1-12)</td>
</tr>
<tr>
<td>MONTH_NAME</td>
<td>VARCHAR2(50)</td>
<td>Month name (January-December)</td>
</tr>
<tr>
<td>DAY_OF_MONTH</td>
<td>NUMBER(10)</td>
<td>Day of the month (1-31)</td>
</tr>
<tr>
<td>DAY_OF_WEEK</td>
<td>NUMBER(10)</td>
<td>Day of the week (1-7)</td>
</tr>
<tr>
<td>DAY_OF_WEEK_NAME</td>
<td>VARCHAR2(50)</td>
<td>Day (Sunday-Saturday)</td>
</tr>
<tr>
<td>WEEK_OF_YEAR</td>
<td>NUMBER(10)</td>
<td>Week of the year (1-52)</td>
</tr>
<tr>
<td>HOUR</td>
<td>NUMBER(10)</td>
<td>Hour of the day (0-23)</td>
</tr>
</tbody>
</table>
Usage_Metric

Usage_Metric table is a multi-dimensional OLAP fact table for capturing the demand of Dependency, Service, and Network Element. The granularity of the data is of a single hour. Together with the time table and the related entities, it is sliced and diced as needed. Table 4-18 lists the contents of the Usage_Metric table.

Table 4-18. Usage_Metric Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENTITY_ID</td>
<td>NUMBER(19)</td>
<td>The data contains foreign key to entity types: Network Element, Service, and Dependency</td>
</tr>
<tr>
<td>TIME_ID</td>
<td>NUMBER(19)</td>
<td>Time identifier (foreign key)</td>
</tr>
<tr>
<td>BASELINE_TIME_ID</td>
<td>NUMBER(19)</td>
<td>Baseline time identifier contains the hour part of the TIME_ID. The baseline is for 24 hours (foreign key)</td>
</tr>
<tr>
<td>VALUE</td>
<td>NUMBER(19)</td>
<td>The number of network transactions in the given time (hour granularity)</td>
</tr>
</tbody>
</table>

Usage_Metric_Baseline

Usage_Metric_Baseline table describes the baseline usage composed of the entity’s usage metric accumulated within the first 24 hours after it is detected by the system. Table 4-19 lists the contents of the Usage_Metric_Baseline table.

Table 4-19. Usage_Metric_Baseline Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASELINE_TIME_ID</td>
<td>NUMBER(19)</td>
<td>Hour part of the TIME_ID (primary key)</td>
</tr>
<tr>
<td>ENTITY_ID</td>
<td>NUMBER(19)</td>
<td>The data contains foreign key to entity types: Network Element, Service, and Dependency</td>
</tr>
<tr>
<td>VALUE</td>
<td>NUMBER(19)</td>
<td>Number of network transactions</td>
</tr>
</tbody>
</table>

Virtual_System_Settings

Virtual_System_Settings table represents the virtual system settings on a specific virtual container machine. Table 4-20 lists the contents of the Virtual_System_Settings table (Introduced in version 6.0 of ADM).

Table 4-20. Virtual_System_Settings Table

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>NUMBER(19)</td>
<td>Unique identifier for each record of the Virtual_System_Settings table (primary key)</td>
</tr>
<tr>
<td>CONTAINER_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier on which the current virtual system settings reside (foreign key)</td>
</tr>
<tr>
<td>UUID</td>
<td>VARCHAR2(256)</td>
<td>Unique identifier of the virtual host</td>
</tr>
<tr>
<td>STATE</td>
<td>VARCHAR2(256)</td>
<td>State of the virtual host</td>
</tr>
<tr>
<td>VM_NAME</td>
<td>VARCHAR2(256)</td>
<td>Name of the virtual host</td>
</tr>
<tr>
<td>OS_FAMILY</td>
<td>VARCHAR2(256)</td>
<td>Operating system family of the virtual host</td>
</tr>
<tr>
<td>OS_NAME</td>
<td>VARCHAR2(256)</td>
<td>Operating system name of the virtual host</td>
</tr>
<tr>
<td>OS_TYPE</td>
<td>VARCHAR2(256)</td>
<td>Operating system type of the virtual host</td>
</tr>
<tr>
<td>TOOLS_STATUS</td>
<td>VARCHAR2(256)</td>
<td>VMware tools status</td>
</tr>
<tr>
<td>TOOLS_VERSION</td>
<td>VARCHAR2(256)</td>
<td>VMware tools version</td>
</tr>
<tr>
<td>TOTAL_MEM_SIZE</td>
<td>NUMBER(19)</td>
<td>Total memory of the virtual host</td>
</tr>
</tbody>
</table>
### Table 4-20. Virtual_System_Settings Table (Continued)

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPS</td>
<td>VARCHAR2(1024)</td>
<td>List of IP addresses that are available on the virtual host</td>
</tr>
<tr>
<td>MACS</td>
<td>VARCHAR2(4000)</td>
<td>List of MAC addresses that are available on the virtual host</td>
</tr>
<tr>
<td>VIRTUAL_NE_ID</td>
<td>NUMBER(19)</td>
<td>Network element identifier of the virtual host connected on the other side (foreign key)</td>
</tr>
</tbody>
</table>
This chapter provides information on “Creating Custom Reports” on page 41.

Creating Custom Reports

Most reports are generated from the four core tables demonstrated in Figure 4-1. The reports cover inventory, dependency, and usage. The rest of the data is joined for special reports without much complication. The following are examples of reports that are generated from the ADM external database.

Host Inventory

The following script generates a report of all hosts that were discovered between 4:54 P.M., October 29, 2006 and 5:04 P.M., October 13, 2007:

```
select ne.host_ips, ne.hostnames, ne.groups, ne.discovery_time
from network_element ne
where discovery_time between to_date('2006-29-10 16:54:00', 'yyyy-dd-mm hh24:mi:ss') and to_date('2007-13-10 17:04:00', 'yyyy-dd-mm hh24:mi:ss')
```

Connection Inventory

For the specified type of connection, source service, and IP addresses, the report displays the connection type, version, source service, source host, port, IP address, and discovery date:

```
select dep.protocol, dep.groups, dep.src_service_name, dep.src_hostnames, dep.src_port, dep.trg_service_name, dep.trg_hostnames, dep.trg_port, dep.discovery_time
from dependency dep
where discovery_time between to_date('2006-29-10 16:54:00', 'yyyy-dd-mm hh24:mi:ss') and to_date('2007-13-10 17:04:00', 'yyyy-dd-mm hh24:mi:ss')
```

Most Used Services

This report list the 10 most used services used for a selected date and time:

```
select * from
(select service.primary_host_ip, service.primary_hostname, service.name, service.version, service.vendor, service.category, max(value)as peak, sum(value)as transactions
from usage_metric metric inner join service service on service.id = metric.entity_id
where metric.time_id between 2006102916 and 2007101317 and service.name is not null
group by service.name, service.version, service.vendor, service.category, service.primary_hostname, service.primary_host_ip
) where rownum <= 10
```
**Application Dependency**

For the specified Business Application, this report shows each host and service the application depends on and each host and service that depends on it (if any):

```sql
select distinct 0 as degree, grp.group_name, service.name, service.version, service.vendor,
    service.category, service.primary_host_ip, service.primary_hostname,
    service.discovery_time
from dependency dep inner join group_membership grp on grp.entity_id = dep.id
inner join service service on service.id =dep.src_service_id
where service.name is not null and grp.group_name = '111'
union
select distinct 1 as degree, grp.group_name, service.name, service.version, service.vendor,
    service.category,
    service.primary_host_ip, service.primary_hostname, service.discovery_time
from dependency dep inner join group_membership grp on grp.entity_id = dep.id
inner join service service on service.id =dep.trg_service_id
where service.name is not null and grp.group_name = '111'
```
Index

A
ADM components 7

C
configuration 16
   for MS SQL 21
   for Oracle 18
custom reports 41

D
database dependencies 28
database structure 28

E
ERD schema 27
external database 28
external reports 11

F
flat file creation 9
foreign key 28
FTP 8, 11

H
hardware requirements 15

I
installation 16
   for MS SQL 20
   for Oracle 16

L
loading data to the external database 23

M
MS SQL as the external database 16

O
Oracle as the external database 15

P
policy 8
primary key 29

R
report 8

S
scheduling 12
SFTP 12
software requirements 15
status of the load process 23
synchronizing the external database 8

T
table join 28

U
upgrading the external database 24
   for MS SQL 24
   for Oracle 25