CA Identity Manager

Installation Guide (WebSphere)

r12.5 SP6

CA Product References

This document references the following CA products:

- CA Identity Manager
- CA SiteMinder®
- CA Directory
- CA Enterprise Log Manager
- CA Role & Compliance Manager

Contact CA

Contact CA Support

For your convenience, CA provides one site where you can access the information you need for your Home Office, Small Business, and Enterprise CA products. At http://ca.com/support, you can access the following:

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

Provide Feedback

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

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

Contents

Chapter 1: Installation Overview

This guide provides instructions for installing CA Identity Manager and also includes information on optional components for installation such as Provisioning and CA SiteMinder.

This section contains the following topics:

Sample CA Identity Manager Installations

Based on the functionality you want to implement, you can select which components of CA Identity Manager you want to install in your environment.

In all CA Identity Manager installations, the Identity Manager Server is installed on an application server. After you install the application server, you use the CA Identity Manager Installer to install the software you need. The following sections illustrate some examples of CA Identity Manager implementations at a high level.

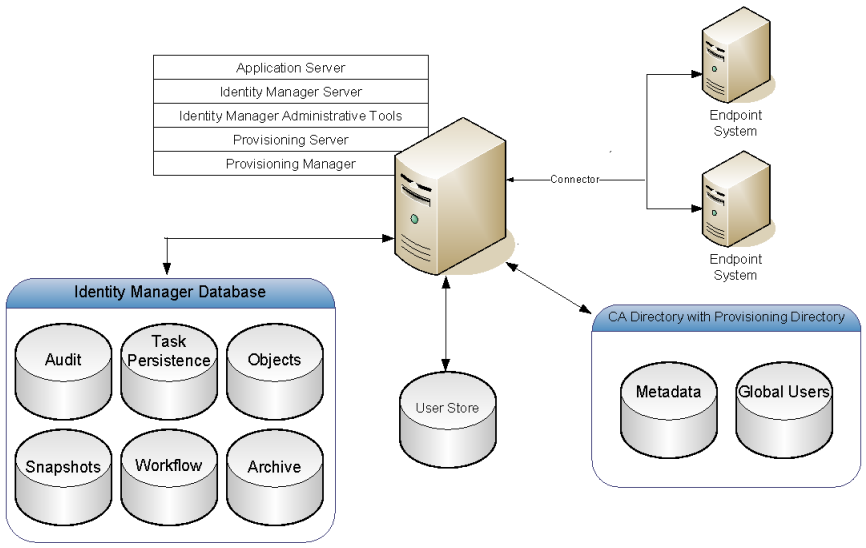
Basic Installation

In a basic installation, all software components are installed on the same system. Two types of basic installation exist:

- A standalone installation -- all software is on one system, suitable for product demonstration
- A distributed installation -- one copy of each component is installed, but components are on different systems

CA Identity Manager Provisioning allows you to create an Environment that connects to a Provisioning Server for provisioning accounts to various endpoint systems. You can assign provisioning roles to users you create through CA Identity Manager. Provisioning roles are associated with account templates that define accounts that users can receive on endpoint systems. Account templates provide users with access to additional resources, such as an email account.

The accounts exist in managed endpoints defined by the account templates. The following figure is an example of a basic CA Identity Manager installation with Provisioning:



Identity Manager Server

Executes tasks within CA Identity Manager. The J2EE Identity Manager application includes the Management Console (for configuring environments), and the User Console (for managing an environment).

Identity Manager Administrative Tools

Provides tools and samples for configuring and using CA Identity Manager. The tools include configuration files, scripts, utilities, and jar files that you use to compile custom objects with CA Identity Manager APIs and API samples. The Provisioning Manager and WorkPoint Designer are also included with the Administrative Tools.

The default installation location for most Administrative Tools follows:

- **Windows:** C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools
- **UNIX:** /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools

However, the default location for Provisioning Manager, which is only installed on Windows, follows:

- C:\Program Files\CA\Identity Manager\Provisioning Manager

Identity Manager Database

Stores data for CA Identity Manager. This database stores information for auditing, task persistence, snapshots (reporting), workflow, and Identity Manager objects. This database must be a relational database.

Note: For a complete list of supported relational databases, see the CA Identity Manager support matrix on the CA Support Site.

Identity Manager User Store

Contains users and their information. This store can be a pre-existing user store already in use by the company. This user store can be LDAP or a relational database.

Note: For more information about setting up a user store for CA Identity Manager, see the *Configuration Guide*.

Identity Manager Provisioning Server

Manages accounts on endpoint systems. On the same system or another system, you can also install Connector Servers, which manage Java or C++ based connectors to endpoints.

Identity Manager Provisioning Directory

Specifies the Provisioning Directory schema to CA Directory. This schema sets up the Directory System Agents (DSAs) within CA Directory. The Identity Manager user store can also be the Provisioning Directory.

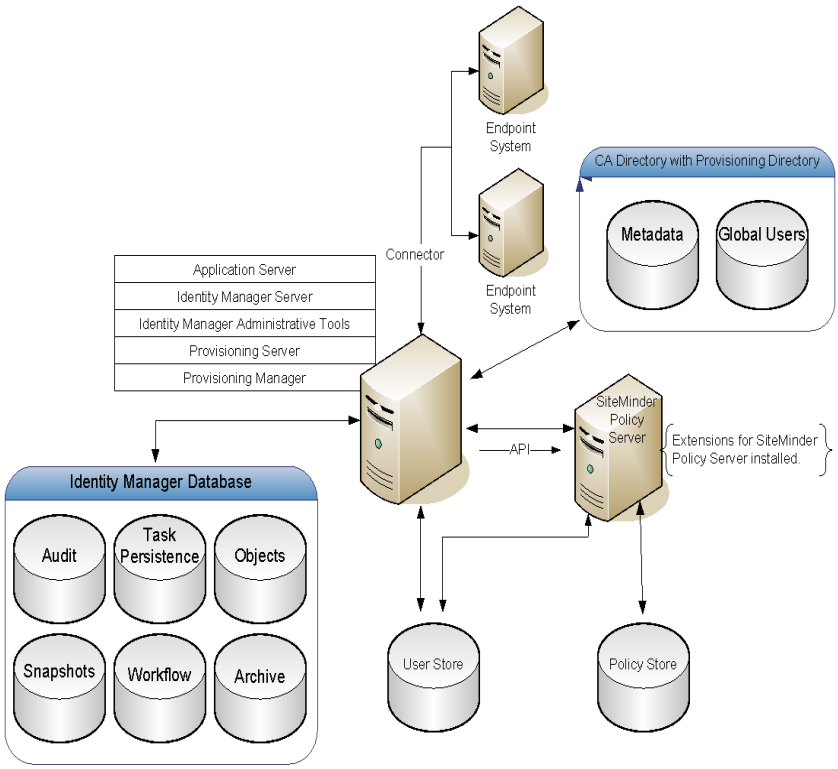
Identity Manager Provisioning Manager

Manages the Provisioning Server through a graphical interface. This tool is used for administrative tasks such as synchronizing accounts with account templates. The Provisioning Manager is installed as part of the Identity Manager Administrative Tools or can be installed separately from those tools.

Note: This application runs on Windows only.

Installation with a SiteMinder Policy Server

CA Identity Manager can be integrated with a SiteMinder Policy Server, which provides advanced authentication and protection for your Environment. The following figure is an example of a CA Identity Manager installation with a CA SiteMinder Web Access Manager Policy Server:



Identity Manager Server

Executes tasks within CA Identity Manager. The J2EE Identity Manager application includes the Management Console (for configuring environments), and the User Console (for managing an environment).

Identity Manager Administrative Tools

Provides tools and samples for configuring and using CA Identity Manager. The tools include configuration files, scripts, utilities, and jar files that you use to compile custom objects with CA Identity Manager APIs and API samples. The Provisioning Manager and WorkPoint Designer are also included with the Administrative Tools.

The default installation location for most Administrative Tools follows:

- **Windows:** C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools
- **UNIX:** /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools

However, the default location for Provisioning Manager, which is only installed on Windows, follows:

- C:\Program Files\CA\Identity Manager\Provisioning Manager

Identity Manager Database

Stores data for CA Identity Manager. This database stores information for auditing, task persistence, snapshots (reporting), workflow, and Identity Manager objects. This database must be a relational database.

Note: For a complete list of supported relational databases, see the CA Identity Manager support matrix on the CA Support Site.

Identity Manager User Store

Contains users and their information. This store can be a pre-existing user store already in use by the company. This user store can be LDAP or a relational database.

Note: For more information about setting up a user store for CA Identity Manager, see the *Configuration Guide*.

SiteMinder Web Agent

Works with the SiteMinder Policy Server to protect the User Console. Installed on the system with the Identity Manager Server.

SiteMinder Policy Server

Provides advanced authentication and authorization for CA Identity Manager and facilities such as Password Services, and Single Sign-On.

Extensions for SiteMinder Policy Server

Enable a SiteMinder Policy Server to support CA Identity Manager. Install the extensions on each SiteMinder Policy Server system in your CA Identity Manager implementation.

High Availability Installation

Before installing CA Identity Manager, consider what your goals are. For example, you may want a resilient implementation that consistently provides good performance. You may also want to make the implementation scalable, so you can easily add users and systems over many different network operating systems, security systems, databases, and groupware products.

A high-availability implementation provides the following features:

- Failover—Switches to another system automatically if the primary system fails or is temporarily offline for any reason
- Load balancing—Distributes processing and communications activity evenly across a computer network so that performance remains good and no single device is overwhelmed
- Various deployment tiers that provide the flexibility to serve dynamic business requirements

A high-availability implementation addresses the following requirements:

- The Identity Manager Server can be installed on an application server to allow failover to any of the nodes in the cluster, providing uninterrupted access to users.
- The Provisioning Directory uses a CA Directory router to route Provisioning Server directory traffic using the X.500 protocol.
- CA Identity Manager includes the connector servers that can be configured per-directory or per-managed systems. Installing multiple connector servers increases resilience. Each connector server is also an LDAP server, similar to the Provisioning Server.

Identity Manager Server Architecture

An Identity Manager implementation may span a multi-tiered environment that includes a combination of hardware and software, including three tiers:

- Web Server tier
- Application Server tier
- Policy Server tier (optional)

Each tier may contain a cluster of servers that perform the same function to share the workload for that tier. You configure each cluster separately, so that you can add servers only where they are needed. For example, in a clustered Identity Manager implementation, a group of several systems may all have an Identity Manager Server installed. These systems share the work that is performed by the Identity Manager Server.

Note: Nodes from different clusters may exist on the same system. For example, an application server node can be installed on the same system as a Policy Server node.

Provisioning Components Architecture

Provisioning provides high availability solutions in the following three tiers:

- Client tier

  The clients are the Identity Manager User Console, Identity Manager Management Console and the Provisioning Manager. You can group clients together based on their geographic locations, organizational units, business functions, security requirements, provisioning workload, or other administration needs. Generally, we recommend keeping clients close to the endpoints they manage.

- Provisioning Server tier

  Clients use primary and alternate Provisioning Servers, in order of their failover preference. Client requests continue to be submitted to the first server until that server fails, that is, the connection stays active until the server fails. In case of a failure, the client checks the list of configured servers, in order of preference, to find the next available server.

  The Provisioning Server can have multiple connector servers in operation. Each connector server handles operations on a distinct set of endpoints. Therefore, your organization may choose to deploy connector servers on systems that are close in the network to the endpoints. For example, if you have many UNIX /etc endpoints, you might have one connector server installed on each of these servers so that each connector server controls only the endpoint on the server where it is installed.

  Installing Connector Servers close to the endpoints also reduces the delays in managing accounts on those endpoints.

- CA Directory Repository tier (Provisioning Directory)

  You can use another CA Directory router to send server requests to Provisioning Directories. You can replicate multiple Provisioning Directories for load-balancing, failover, or both.

Overall Installation Process

To install CA Identity Manager, perform the following steps:

1. Install the prerequisite hardware and software and configure your system as required.
2. Install the CA Identity Manager components on one system or several systems or install the Identity Manager Server on an application server cluster.

3. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers.
4. (Optional) Install optional provisioning components.
5. (Optional) Protect CA Identity Manager with SiteMinder.
6. (Optional) Install the report server.

Note: In this document, each chapter includes a checklist of the steps to install or configure a CA Identity Manager feature or component. It is the section that begins with a How To title in each chapter. The appendix **Installation Checklists** includes all checklists. Print this appendix before you begin the installation.

Installation Worksheet

During CA Identity Manager installation, you are prompted for the location of software, administrator account names, and other information. To simplify the installation process, complete the **Installation Worksheets** section of the Product Prequisites chapter to have answers ready for these questions.

Chapter 2: Installation Prerequisites

This section contains the following topics:

Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
X	**1. Install prerequisite hardware and software and configure your system as required.**
	2. Perform one of these installations: ■ Basic installation ■ Installation on an application server cluster
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5. (Optional) Protect CA Identity Manager with SiteMinder.
	6.(Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

Prerequisite Knowledge

This guide is intended for users who are familiar with Java, J2EE standards, and application server technology. It assumes that you have the following technical knowledge:

- An understanding of J2EE application servers and multi-tier architecture
- Experience with managing the application server, including tasks such as starting the application server
- Experience with managing a relational database
- (Optional) Familiarity with SiteMinder concepts, terms, and Policy Server configuration tasks

How to Install Prerequisite Components

To install the prerequisite hardware and software for CA Identity Manager:

✓	Step
	1. Make your system meet the hardware and software requirements.
	2. Create a database.
	3. Set up the application server as required.
	4. Fill in the Installation Worksheets with information you need to supply during the CA Identity Manager installation.

Meet System Requirements

Before installing CA Identity Manager, make sure your systems have the right hardware, software, and configuration required.

Check Hardware Requirements

Identity Manager Server

These requirements take into account the requirements of the application server installed on the system where you install the Identity Manager Server.

Component	Minimum	Recommended
CPU	Intel (or compatible) 1.5 GHz (Windows or Red Hat Linux), SPARC 1.0 GHz (Solaris) or POWER4 1.1 GHz (AIX)	Dual core Intel (or compatible) 2.5 GHz (Windows or Red Hat Linux), Dual core SPARC 1.5 GHz (Solaris) POWER5 1.5 GHz (AIX)
Memory	2 GB	4 GB
Available Disk Space	2 GB	2 GB
Temp Space	2 GB	2 GB

Provisioning Server or a Standalone Connector Server

Component	Minimum	Recommended
CPU	Intel (or compatible) 1.5 GHz (Windows) SPARC 1.0 GHz (Solaris)	Dual core Intel (or compatible) 2.5 GHz (Windows) SPARC 1.5 GHz (Solaris)
Memory	2 GB	4 GB
Available Disk Space	2 GB	2 GB

Provisioning Directory

Component	Minimum	Recommended
CPU	Intel (or compatible) 1.5 GHz (Windows) SPARC 1.0 GHz (Solaris)	Dual core Intel (or compatible) 2.5 GHz (Windows) SPARC 1.5 GHz (Solaris)
Memory	2 GB	4 GB

Component	Minimum	Recommended
Available Disk Space	2 to 10 GB, depending on the number of endpoint accounts ■ Compact—Up to 10,000 accounts, 0.25 GB per datafile (total 1 GB) ■ Basic—Up to 400,000 accounts, 0.5 GB per datafile, (total 2 GB) ■ Intermediate (64 bit only)—Up to 600,000 accounts, 1 GB per datafile, total 4 GB ■ Large (64 bit only)—Over 600,000 accounts, 2 GB per datafile, total 8 GB	2 to 10 GB, depending on the number of endpoint accounts ■ Compact—Up to 10,000 accounts, 0.25 GB per datafile (total 1 GB) ■ Basic—Up to 400,000 accounts, 0.5 GB per datafile, (total 2 GB) ■ Intermediate (64 bit only)—Up to 600,000 accounts, 1 GB per datafile, total 4 GB ■ Large (64 bit only)—Over 600,000 accounts, 2 GB per datafile, total 8 GB
Processor	32-bit processor and operating system for small deployments 64-bit processor and operating system for intermediate and large deployments	64-bit processor and operating system

All Components on One System

Hosting the entire CA Identity Manager product on a single physical system is not recommended for production environments. However, to do so, the hardware requirements are as follows:

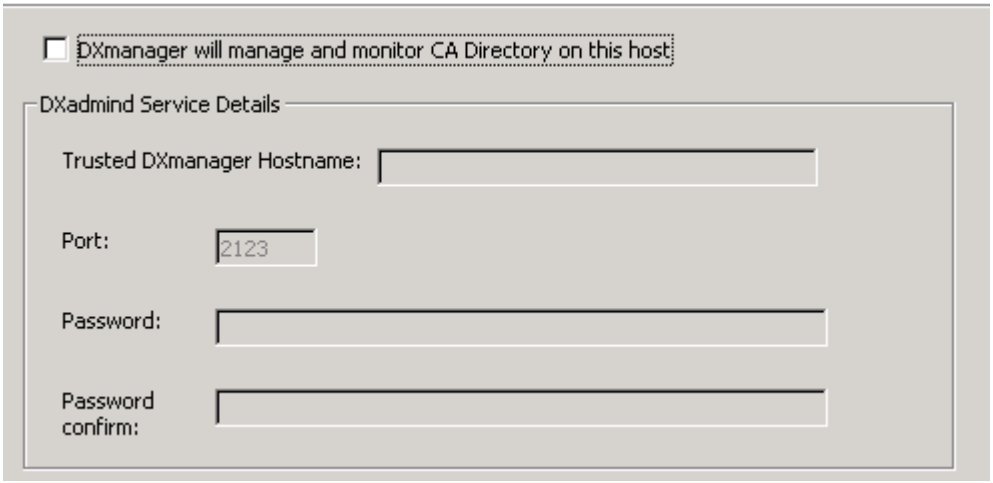
Component	Minimum
CPU	Intel (or compatible) 2.0 GHz (Windows) SPARC 1.5 GHz (Solaris)
Memory	4 GB
Available Disk Space	6 to 14 GB depending on the number of accounts
Processor	64 bit processor and operating system for intermediate and large deployments

Install CA Directory

Before you install CA Identity Manager, install CA Directory using the following steps:

1. Install CA Directory on the system where you plan to install the Provisioning Directory. A supported version of CA Directory is included on your installation media. For details on installation, download the CA Directory documentation from the support site.

   Note: When the installer asks about installing DXadmind for DXManager, you can safely uncheck this option. The Provisioning Directory does not use DXManager.



2. Install a second copy of CA Directory on the system where you plan to install the Provisioning Server. This installation is for routing purposes, so that the Provisioning Server can communicate with the remote Provisioning Directory.

Important! We recommend that you disable all antivirus software before installation. If antivirus software is enabled while installation takes place, problems can occur. Remember to re-enable your antivirus protection after you complete the installation.

Create a FIPS 140-2 Encryption Key

When you run the CA Identity Manager installer, you are given the option of enabling FIPS 140-2 compliance mode. For CA Identity Manager to support FIPS 140-2, all components in a CA Identity Manager environment must be FIPS 140-2 enabled. You need a FIPS encryption key to enable FIPS 140-2 during installation. A Password Tool for creating a FIPS key is located in the installation media at PasswordTool\bin.

Important! Use the same FIPS 140-2 encryption key in all installations and be sure that you safeguard the key file once generated by the Password Tool.

(Optional) Integrate with SiteMinder

A SiteMinder Policy Server is an optional component that you install as described in the *SiteMinder Installation Guide*. If you plan to make the policy server highly available, you configure it as a policy server cluster.

To install a policy server

1. Install the SiteMinder Policy Server. For details, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide*.
2. If you plan to make the policy server highly available, install it on each node that should be in the Policy Server cluster.

   Note: Each Policy Server in the cluster uses the same policy store.
3. Check that you can ping the systems that host the Policy Server from the system where you plan to install the Identity Manager Server.

To install the Identity Manager Extensions for SiteMinder

Before installing the Identity Manager server, you add the extensions to each Policy Server. If the Policy Server is on the system where you plan to install the Identity Manager server, you can install the extensions and the Identity Manager server simultaneously. If so, omit this procedure.

1. Stop the SiteMinder services.
2. Install the Identity Manager Extensions for SiteMinder. Do one of the following:
   - **Windows:** From your installation media, run the following program in the top-level folder:
     `ca-im-r12.5sp*N*-win32.exe`
   - **UNIX:** From your installation media, run the following program in the top-level folder:
     `ca-im-r12.5sp*N*-sol.bin`

   sp*N* represents the current SP release of CA Identity Manager.
3. Select Extensions for SiteMinder.
4. Complete the instructions in the installation dialog boxes.

Create the Database

CA Identity Manager requires a relational database to store objects and data for auditing, snapshots (reporting), workflow, and task persistence. Install a supported version of Oracle or Microsoft SQL Server and create a database.

When installing CA Identity Manager, all of the database schemas are created automatically when the application server is started. However, after installing CA Identity Manager, you can configure separate databases for auditing, snapshots (reporting), workflow, and task persistence. To create these databases, see the chapter on Separate Database Configuration.

WebSphere Application Server

When using WebSphere as the CA Identity Manager application server, note the following:

- The Identity Manager Server is a J2EE application that is deployed on a supported application server.
- If you are installing CA Identity Manager on Solaris, run the installation as root.
- When using WebSphere on Windows, be sure that your Admin username is less than 12 characters long. If you have a username that is 12 characters or greater, CA Identity Manager will not work. For example, the username "Administrator" is greater than 12 characters and will cause CA Identity Manager to fail.
- Be sure to install WebSphere in a directory pathname that contains no spaces.
- The Application Server connects to the Provisioning Server and other servers by SSL. See the Application Server documentation for information on configuring SSL, including information on certificates and keys.

Install a WebSphere Application Server

Install a supported version of WebSphere and the JRE or JDK it requires.

Note: For a complete list of supported platforms and versions, see the CA Identity Manager support matrix on CA Support.

1. To use IBM WebSphere as the application server for CA Identity Manager, install the WebSphere server as described in IBM's documentation.

   Select the following components during the installation:

   - The appropriate plug-in for your Web Server.

- WebSphere 6.1 uses default messaging.
- For all WebSphere versions, be sure to select the Server and Client option.

Important! We recommend that you disable security at profile creation. For example, create a WebSphere 6.1 profile with the Security Enabled option unchecked.

2. Install the required version JDK or JRE before installing the Identity Manager Server.

Verify WebSphere

Use the following tests to verify that WebSphere is working:

- Test whether the WebSphere application server is installed correctly by accessing IBM's snoop utility at the following URL:

  http://*hostname*:*port*/snoop

  For example:

  http://MyServer.MyCompany.com:9080/snoop

  If WebSphere is installed correctly, the Snoop Servlet—Request Client Information page is displayed in the browser.

- If you have a web server installed, test that the WebSphere application server plug-in is installed correctly. Use IBM's snoop utility without including the application server port in the URL:

  http://*hostname*/snoop

  For example:

  http://MyServer.MyCompany.com/snoop

  If WebSphere is installed correctly, the same Snoop Servlet—Request Client Information page is displayed in the browser. This means that profile was created and has at least one server which is configured with the plug-in.

For additional help with WebSphere, contact IBM customer support.

Configure WebSphere for CA Identity Manager

Perform the following steps to ensure that your CA Identity Manager installation succeeds on WebSphere.

1. Save any changes to the WebSphere configuration via the Admin Console (Save to Master Configuration).
2. Shut down the application server.

3. Remove the contents of the following folders:
   - Temp Directory:
     - Windows: %temp%
     - Unix: /tmp/*
   - *Websphere_home*/profiles/WAS_PROFILE/temp/*
   - *Websphere_home*/profiles/WAS_PROFILE/wstemp/*
   - *Websphere_home*/profiles/WAS_PROFILE/tranlog/*
   - *Websphere_home*/profiles/WAS_PROFILE/configuration/*
   - *Websphere_home*/deploytool/itp/configuration/org.*, leaving only config.ini in this directory
4. In the *Websphere_home*/profiles/WAS_PROFILE/properties/soap.client.props file, set com.ibm.SOAP.requestTimeout to 1800 or higher.

Note: For more information, see your WebSphere documentation.

Important! Restart your WebSphere application server before starting the CA Identity Manager installation.

Enable XA Transactions for Microsoft SQL Server

If you are using WebSphere with Microsoft SQL Server, enable XA transactions on Microsoft SQL Server. CA Identity Manager needs an XA data source for the database transactions to work properly.

To enable XA Transactions for Microsoft SQL Server

1. Download the SQL Server JDBC Driver version 1.2 from Microsoft.

   Note: The download may first present an HTM file that is a license agreement for you to approve.

2. Open the default help in the downloaded SQL Server Driver package.
3. Search for the **XA transactions** in the help.
4. In the Understanding XA Transactions, perform the following procedures:
   - Running the MS DTC Service
   - Configuring the JDBC Distributed Transaction Components

   In performing these procedures, verify the following are true:
   - When you run the xa_install.sql script, make sure you get a script complete message. You can ignore the drop table errors, which appear the first time that you run the script.
   - When you add the user to the SqlJDBCXAUser role, that user needs to be in the master database.

Complete the Installation Worksheets

The CA Identity Manager installation program asks you for information about previously installed software and the software that you are installing. If you are running the CA Identity Manager installer on an IPv6 system, ensure that you provide hostnames (and not IP addresses) in the installer screens.

Note: Use the following **Installation Worksheet** to record this information. We recommend that you complete the worksheet before starting the installation.

Provisioning Directory

Record the following Provisioning Directory and Provisioning Server information you need during the CA Identity Manager installation.

Field Name	Description	Your Response
Provisioning Directory Host	The hostname of the Provisioning Directory system if it is remote. You need the hostnames for the primary and any alternate Provisioning Directories.	
Shared Secret	The special password for the Provisioning Directory. Use the same password for the primary and any alternate Provisioning Directories.	
Provisioning Server Hostname	The host names of the primary and any alternate Provisioning Servers.	

WebSphere Information

Record the following WebSphere information you need during the CA Identity Manager installation:

Field Name	Description	Your Response
WebSphere Install Folder	The location of the application server home directory.	
Server Name	The name of the system on which the application server is running.	
Profile Name	The name of the profile you want to use for CA Identity Manager.	
Cell Name	The name of the cell in which the application server is located.	
Node Name	The name of the node in which the application server is located.	

Field Name	Description	Your Response
Cluster Name	The cluster name for high-availability implementations. This is only needed if you plan on installing CA Identity Manager in a clustered environment.	
App Server URL and port	The application URL and port number of the system that will host the Identity Manager Server (system that will host the application server).	

Database Connection Information

An Oracle or Microsoft SQL Server database must already be configured and working. Record the following database information you need during the CA Identity Manager installation:

Field Name	Description	Your Response
Database Type	The database type (vendor/version) of the database created for task persistence, workflow, audit, reporting, object storage, and task persistence archive.	
Host Name	The hostname of the system where the database is located. **Note:** Ensure you provide a hostname and *not* an IP address.	
Port Number	The port number of the database.	
Database Name	The database identifier.	
Username	The username for database access. **Note:** This user must have administrative rights to the database unless you plan to import the schema manually.	
Password	The password for the user account with administrative rights.	

Login Information

Record the following passwords you need during the Provisioning Components installation.

Field Name	Description	Your Response
Username	A username that you create to log into the provisioning components.	
Provisioning Server password	A password for this Server.	
C++ Connector Server password	A password needed for this server. Each C++ Connector Server can have a unique password.	
Provisioning Directory password	A password used by Provisioning Server to connect to Provisioning Directory. For an alternate Provisioning Server, enter the Provisioning Directory password created for the primary Provisioning Server.	

SiteMinder Information

If you plan to use a SiteMinder Policy Server to protect CA Identity Manager, record the following information:

Field Name	Description	Your Response
Policy Server Host Name	The hostname of the SiteMinder Policy Server.	
SiteMinder Administrator Name	The administrator username for the SiteMinder Policy Server.	
SiteMinder Administrator Password	The administrator user password for the SiteMinder Policy Server.	
SiteMinder Folder (Solaris Only)	The location of SiteMinder on the system with a SiteMinder Policy Server installed.	

Field Name	Description	Your Response
SiteMinder Agent Name	The name of the SiteMinder Agent that CA Identity Manager will use to connect to SiteMinder.	
SiteMinder Shared Secret	The shared secret for the above Agent.	

Chapter 3: Basic Installation

This section contains the following topics:

Installation Status

This table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
X	**2. Perform one of these installations:** ■ **Basic installation** ■ **Installation on an application server cluster**
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5. (Optional) Protect CA Identity Manager with SiteMinder.
	6.(Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

CA Identity Manager Components

A basic installation occurs when you install components on different systems. You install one copy of each component, but use two or more systems for where you install them.

Note: If you intend to install multiple copies of components for high availability, see the chapters on installation on a cluster and high-availability provisioning installation.

Install one of each of the following components on a system at your site:

- Identity Manager Server—The server that provides the core functionality of the product.
- Identity Manager Administrative Tools—Install tools such as the Provisioning Manager, which runs on a Windows system.
- Identity Manager Provisioning Server—Enables provisioning in CA Identity Manager.
- Identity Manager Provisioning Directory Initialization—Configures a directory to store provisioning data. Use the installation program on each system where CA Directory is installed.
- Extensions for SiteMinder—Extend the SiteMinder Policy Server if you are using it to protect CA Identity Manager. Install these extensions on the same system as the Policy Server before you install the Identity Manager Server.

How to Perform a Basic Installation

Use the following checklist to perform a basic installation of CA Identity Manager:

✓	Step
	1. Install CA Identity Manager components on the systems required.
	2. Add support for SiteMinder if you are installing on a 64-bit AIX system.
	3. Verify the Identity Manager Server starts.
	4. Configure Provisioning Manager if installed on a remote system.
	5. Install optional provisioning components.

Install CA Identity Manager Components

For a production environment, use separate systems for data servers. For example, we recommend that the Provisioning Directory and a database (SQL or Oracle) are on a separate system from the Identity Manager Server and the Provisioning Server. If you are installing SiteMinder, you may also prefer to have it on a separate system. The Administrative Tools can be installed on any system.

Use the CA Identity Manager installer to perform the installation on the systems required. In the procedures that follow, the step to run the installer refers to this program in your installation media's top-level folder:

- **Windows:**
  `ca-im-release-win32.exe`
- **UNIX:**
  `ca-im-release-sol.bin`

release represents the current release of CA Identity Manager.

For each component that you install, make sure that you have the required information for installer screens, (see page 28) such as host names and passwords. If any issues occur during installation, check the installation logs (see page 143).

To install the Extensions for SiteMinder

1. Log into the system where SiteMinder is installed as a Local Administrator (for Windows) or root (for Solaris).
2. Stop the SiteMinder services.
3. Run the installer and select Extensions for SiteMinder.

To install the Identity Manager Server

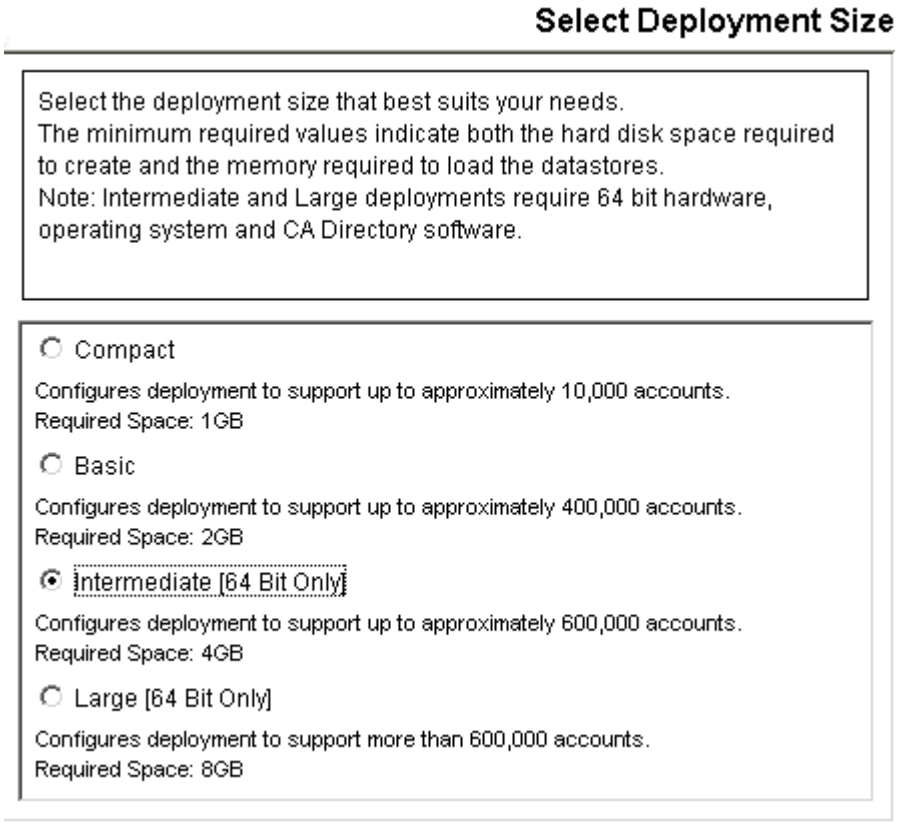
1. If you have installed SiteMinder on a separate system, ensure that you have installed the extensions for SiteMinder there also.
2. Log into the system where the application server is installed as a Local Administrator (for Windows) or root (for Solaris).
3. Stop the application server.
4. Run the installer and select the Identity Manager Server.
5. If you have SiteMinder on the local system, select Extensions for SiteMinder. If it is on a remote system, select Connect to Existing SiteMinder Policy Server.

To install the Provisioning Directory

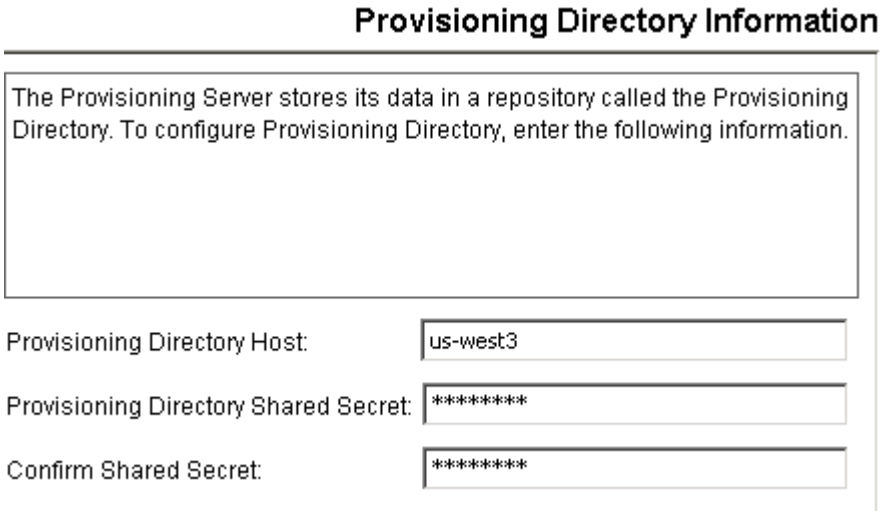
1. Log into the system as a Local Administrator (for Windows) or root (for Solaris).
2. Ensure that CA Directory is already installed on the system.

3. Run the installer and select the Identity Manager Provisioning Directory Initialization.
4. Answer the question about deployment size. Consider the following guidelines, while allowing room for future growth:
   - Compact—up to 10,000 accounts
   - Basic—up to 400,000 accounts
   - Intermediate (64 bit only)—up to 600,000 accounts
   - Large (64 bit only)—more than 600,000 accounts

   Note: If you are installing a Provisioning Directory in an established CA Identity Manager installation, be sure to make the deployment size large enough. Otherwise, an error occurs because the data does not fit when loaded into the data files. Intermediate and Large installations require 64-bit Directory installs (either Solaris or Windows 64 bit).



5. When you enter any password or shared secret in the installation, be sure to provide a password that you can recall when needed.



To install the Provisioning Server

1. Log into the system as a Local Administrator (for Windows) or root (for Solaris).
2. Ensure that CA Directory is already installed and you have the details of the remote Provisioning Directory.
3. Run the installer and select the Identity Manager Provisioning Server.

Add SiteMinder Support on 64-bit AIX

This procedure applies if all of the following conditions are true:

- You are installing CA Identity Manager on AIX 6.1
- You have installed the 64-bit version of Websphere and are using a 64-bit JVM/JRE
- You are integrating CA Identity Manager with Siteminder

Important! If any one of these conditions is false, omit this procedure.

To add SiteMinder Support on 64-bit AIX

In this procedure, you replace the default 32-bit Webagent API libraries included with CA Identity Manager with the appropriate 64-bit libraries from the support site.

Note: The Webagent API libraries are four .so files distributed withCA Identity Manager (libsmagentapi.so, libsmcommonutil.so, libsmerrlog.so, and libsmjavaagentapi.so) and exist in the following location:
WebSphereRoot/IBM/WebSphere/Appserver/profiles/*AppServerName*/installedApps/*CellName*/IdentityMinder.ear/library

1. Download IBM AIX 6.1 64-bit IM WebAgent APIs as follows:
   a. Go to the CA support site (support.ca.com).
   b. Expand the Download Center navigation item on the left-hand side of the page.
   c. Select the Products link.
   d. Locate the first drop down in the grayed out area in the middle of the page, just below All Products.
   e. Select Identity Manager External User - MULTI-PLATFORM.
   f. For Select A Release, choose 12.5.
   g. For Select a Gen Level, choose SP6.
   h. Enable the checkbox called Show me published solutions for this release.
   i. Select CA Identity Manager from the selection box below the checkbox.
   j. Click Go.

2. On the resulting page look for the following:
   `IBM AIX 6.1 64-bit IM WebAgent API-ESD Only (GEN02170809E)`
3. Select Download on the right-hand side.

   On the resulting page, select a download method and download the ZIP file.
4. Expand the download ZIP file. You should see four WebAgent Library files:
   - libsmagentapi.so
   - libsmcommonutil.so
   - libsmerrlog.so
   - libsmjavaagentapi.so
5. Stop the application server.
6. Copy the unzipped 64-bit Webagent Libraries to this location:
   `<WebSphereRoot>/IBM/WebSphere/Appserver/profiles/<AppServerName>/installedApps/<CellName>/IdentityMinder.ear/library`
7. Restart the application server.

Verify the Identity Manager Server Starts

To verify that the Identity Manager Server starts

1. Start CA Identity Manager as follows:
   - **Windows:**

     Navigate to Start, Programs, IBM WebSphere, Application Server 6.x, Profiles, *Profile_Type,* Start the Server

Note: To view status information, use the First Steps console, which you access from the same location as the Start the Server command mentioned above. In the First Steps console, select Start the Server.

- **UNIX:**

  a. Navigate to *websphere_home*/bin from the command line.

  b. Enter the following command:

  startserver *websphere_server*

  When you see a message that resembles the following, the server has completed its startup process:

  `Server server1 is open for e-business`

2. Access the Management Console and confirm the following:

   - You can access the following URL from a browser:

     http://*im_server*:*port*/idmmanage

     For example:

     http://MyServer.MyCompany.com:*port-number*/idmmanage

   - The Management Console opens.
   - No errors are displayed in the application server log.
   - You do not receive an error message when you click the Directories link.

   Note: For details about the Management Console, see the *Configuration Guide*.

Configure a Remote Provisioning Manager

If you installed the Provisioning Manager on a different system from the Provisioning Server, you need to configure communication to the server.

Note: To install the Provisioning Manager, install the Identity Manager Administrative Tools on a Windows system.

To configure a remote Provisioning Manager

1. Log into the Windows system where you installed Provisioning Manager.
2. Go to Start, Programs, CA, Identity Manager, Provisioning Manager Setup.
3. Enter the hostname of the Provisioning Server.
4. Click Configure.
5. For an alternate Provisioning Server, select the domain name from the pull-down list.

6. Click Ok.

   You can now start the Provisioning Manager and see the domain name that you configured.

Install Optional Provisioning Components

Optional Provisioning Components for CA Identity Manager are in the im-pc-*release*.zip

release represents the current release of CA Identity Manager.

The ZIP files includes the following:

SPML Manager

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

SPML Service

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

Remote Agents

Run the specific agent installer from the Provisioning Component media (under \RemoteAgent) to install these components. If you want IPv6 support, you will need to install your agents.

Password Sync Agents

Run the Password Sync Agent installer from the Provisioning Component media (under \Agent) to install this component.

GINA

Run the GINA installer from the Provisioning Component media (under \Agent) to install this component.

Vista Credential Provider

Run the Vista Credential Provider installer from the Provisioning Component media (under \Agent) to install this component.

Bulk Loader Client/PeopleSoft Feed

Run the Bulk Loader Client installer from the Provisioning Component media (under \Clients) to install this component.

JCS SDK

Run the JCS SDK installer from the CA Identity Manager media (under \Provisioning) to install this component.

CCI Standalone

Run the CCI Standalone installer from the Provisioning Component media (under \Infrastructure) to install this component.

More information exists for these components in the following guides:

- Credential Provider (*Administration Guide)*
- GINA Option for Password Reset/Unlock (*Administration Guide)*
- Password Synchronization Agent (*Administration Guide)*
- Connector Xpress (*Connector Xpress Guide)*
- SPML Service (*Provisioning Reference Guide)*
- Agents for use with connectors (*Connectors Guide)*

Connector Xpress

To create your own connectors, you use Connector Xpress to create connectors without expertise required to use a programming interface.

Connector Xpress is a CA Identity Manager utility for managing dynamic connectors, mapping dynamic connectors to endpoints, and establishing routing rules for endpoints. You can use it to configure dynamic connectors to allow provisioning and management of SQL databases and LDAP directories.

Note: For more information on using Connector Xpress, see the *Connector Xpress Guide.*

Connectors

The Identity Manager installer installs all connectors by default. However, in some cases, you must install an agent on an endpoint system you are managing before you can use the related connector.

Connectors run on the Provisioning Server and communicate with the systems managed by an endpoint. For example, systems running Active Directory Services (ADS) can be managed only if the ADS Connector is installed on the Provisioning Server.

Note: For more information about each connector, see the *Connectors Guide*.

Chapter 4: Installation on a WebSphere Cluster

This section contains the following topics:

Installation Status

This table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
X	**2. Perform one of these installations:** ■ **Basic installation** ■ **Installation on an application server cluster**
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5. (Optional) Protect CA Identity Manager with SiteMinder.
	6. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

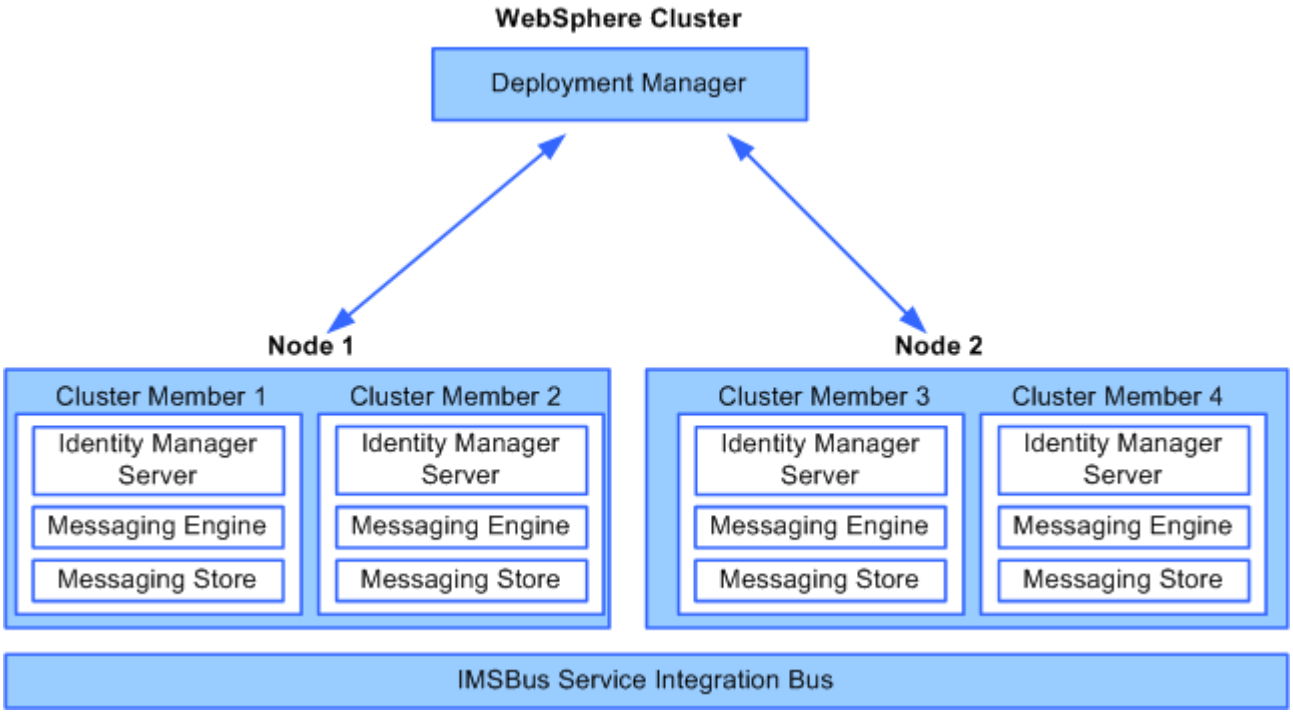
WebSphere Cluster Setup

When you install software for a WebSphere cluster, you set up the following:

- One WebSphere Deployment Manager—Manages the other federated profiles in the cell through node agents.
- One or more nodes—Each node contains one or more cluster members (also called servers), which run the Identity Manager Server.
- Node agent—A process that manages communication between the Deployment Manager and the federated profile.
- Service Integration Bus—Groups resources in WebSphere to simplify administration. The WebSphere cluster is added as a member of the bus.
- Messaging Engine—Provides messaging functionality for members of the service integration bus. One message engine exists for each cluster member.
- Message Store—Stores messages and transaction status for the message engine. Each message engine requires a message store.
- A Web Server—Distributes the load to the appropriate server and, if SiteMinder is installed, protects access to the cluster members.

The following figure shows the relationship between the Deployment Manager, nodes, and cluster members. The Identity Manager Server is installed from the Deployment Manager system to each cluster member, each of which has a messaging engine and a message store.

Note: For more information about these components, see the WebSphere documentation.



WebSphere Cluster Prerequisites

Before you configure CA Identity Manager on a WebSphere cluster, you should be familiar with the concepts and procedures for creating a WebSphere cluster. See the IBM WebSphere documentation for more information about WebSphere clusters.

WebSphere 6.1 Cluster Load Balancing

CA Identity Manager runs on the Service Integration Bus as part of the cluster for WebSphere 6.1 cluster. In this architecture, message-driven beans are bound to the messaging engine on the same host, supporting failover, but not load balancing.

If load balancing is required, for example for the bulk load operations, move the Service Integration Bus out of the cluster and onto dedicated servers. See the WebSphere documentation on Connecting Applications on the Service Integration Bus.

Create Profiles for the Cluster

You set up a WebSphere cluster in the WebSphere Administrator Console.

Note: CA Identity Manager does not support HTTP session persistence in a clustered environment.

To create profiles for the cluster

1. Decide which systems you plan to use for the cluster.
   a. Select a system for the WebSphere Deployment Manager. For best performance, the system should not be used as a node for cluster members.
   b. Determine the cluster member nodes.
2. Install the WebSphere Deployment Manager using the IBM WebSphere documentation for the most recent instructions.

   During the installation, note the directory where you install the Deployment Manager.
   a. Install the IBM WebSphere Application Server Network Deployment software on the Deployment Manager machine.

      When the installation completes, you are prompted to configure a *profile*, a WebSphere runtime environment.
   b. Run the Profile Creation Wizard to create the profile for the Deployment Manager machine. When you are prompted to select a profile type, select the Deployment Manager profile.
   c. Start the Deployment Manager using one of the following methods:
      - Run the StartManager.bat (Windows) or StartManager.sh (Solaris) from a command prompt.

        The *websphere_home*/profiles/*profile_name/*bin folder contains the scripts.
      - If you registered the Deployment Manager as a Windows Service, use Windows Services to start the Deployment Manager.
3. Install the IBM WebSphere Application Server Network Deployment software on each cluster member.
4. Use the Profile Creation Wizard to create a Custom profile for each node.

   A Custom profile allows you to configure a connection to the Deployment Manager.

5. Start each node as follows:
   a. Navigate to *was_home*\WebSphere\AppServer\bin on the system where the managed node is located.
   b. Execute the startNode.bat\.sh command.
6. Confirm that a single cell has all the nodes associated with it at this location:
   was_home/profiles/*Deployment_Manager_Profile*/config/cells/*Cell_Name*/Nodes/
   You should see all federated nodes displayed as folder names.

Creation of profiles may sometimes fail if the bootstrap ports (default: 2809) are not unique. You can check for an error message in the pctLog.txt file in the created profiles' logs folder. For example:

```
(Oct 10, 2007 6:45:55 PM), Install,
com.ibm.ws.install.ni.ismp.actions.ISMPWSProfileLaunchAction, err, INSTCONFFAILED:
Cannot complete required configuration actions after the installation. The
configuration failed. The installation is not successful. Refer to C:\Program
Files\IBM\WebSphere\AppServer\logs\wasprofile\wasprofile_create_CustomIMFromNode.
log for more details.
```

Inspecting the wasprofile_create_CustomIMFromNode.log shows this failure was due to Bootstrap ports that is not unique.

Create the Cluster with One Member

You now configure the cluster with a single member. The other cluster members are added in a subsequent procedure after you install CA Identity Manager.

To create the cluster with one member

1. In the Administrative Console, verify that the nodes show a Synchronized status.
2. Use the Create New Cluster wizard to create the cluster with one member.
   Note the cluster name and the server node name that you create in using this wizard. The server node is the cluster member node.
3. Stop the cluster member, but leave the Node Agents running.
   You may leave the Deployment Manager running.

How to Install CA Identity Manager on a WebSphere Cluster

The following procedures describe how to install CA Identity Manager on a WebSphere cluster.

✓	Step
	1. Run the Installation from the Deployment Manager (see page 49)
	2. Add SiteMinder Support on 64-bit AIX (see page 51)
	3. Export and Import the EAR on 64-Bit AIX Systems (see page 53)
	4. Add Cluster Members (see page 54)
	5. Configure Messaging Engines (see page 54)
	6. Create Message Stores (see page 55)
	7. Create Core Group Policies (see page 56)
	8. Configure Workflow for WebSphere (see page 57)
	9. Update the WebSphere Path for SiteMinder (see page 58)
	10. Configure the Proxy Plug-In for the Web Server (see page 58)

Objects Created by the Installation

You install Identity Manager as described in the following procedure. During the installation, the following EARs are installed on the cluster domain:

- IdentityMinder.ear
- ca-stylesr5.1.1.ear

When you supply a cluster name during the installation, these primary resources are configured:

- Distributed queues/topics targeted to cluster name provided
- Connection factories targeted to server name provided
- Data sources also targeted to cluster name provided
- IMSBus, the Service Integration Bus for CA Identity Manager

Run the Installation from the Deployment Manager

Once you have created the WebSphere cluster, you can install CA Identity Manager on it.

Note: Installer fields that require a hostname and port number should not use localhost.

To install CA Identity Manager on the Deployment Manager server

1. Stop the first cluster member, the only cluster member that you have configured so far.
2. Start the Node Agent for that cluster member.
3. On the system that hosts the WebSphere Deployment Manager, run the CA Identity Manager installation.
   - Windows: From your installation media, run the following program:
     `ca-im-release-win32.exe`
   - UNIX: From your installation media, run the installation program. For example, for Solaris:
     `ca-im-release-sol.bin`

   release represents the current release of CA Identity Manager.

   Important! Make sure that you have the collected the information needed by the installer, such as user names, host names, and ports.
4. Complete the Select Components section by including the Identity Manager Server and any other components that you need on this system.

5. Complete the other sections based on your requirements for the installation.

   The WebSphere section includes these fields:

   WebSphere Install Folder

   The folder or directory where WebSphere is installed. You find this location in the Windows or UNIX file system.

Server Name

The first cluster member in the WebSphere cluster. You find this name in the WebSphere console.

Profile Name

The deployment manager profile. You find this name in the Windows or UNIX file system at the path:

was_home/profiles/*Deployment_Manager_Profile*/config/cells/

Cell Name

The deployment manager's cell which can be found in the WebSphere console.

Node Name

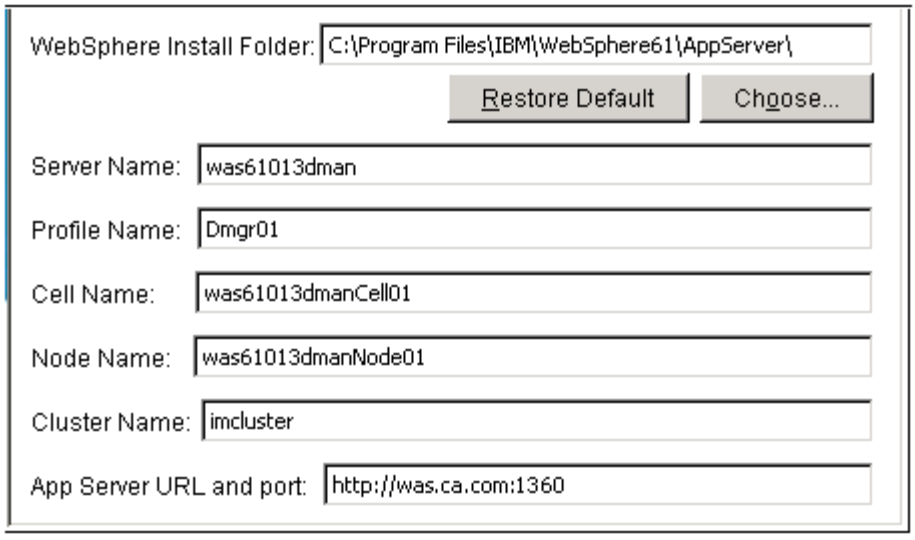
A node that contains the Server Name you supplied on this screen. You find this name in the WebSphere console.

Cluster Name

The name of the cluster. You find this name in the WebSphere console.

App Server URL and port

The URL and port number of the Web Server used for load balancing.



If any issues occur during installation, check the installation logs (see page 143).

Important! Do not start the cluster yet. It will not function at this point. Complete the remaining procedures, which conclude with the steps to start the cluster.

Add SiteMinder Support on 64-bit AIX

This procedure applies if all of the following conditions are true:

- You are installing CA Identity Manager on AIX 6.1
- You have installed the 64-bit version of Websphere and are using a 64-bit JVM/JRE
- You are integrating CA Identity Manager with Siteminder

Important! If any one of these conditions is false, omit this procedure.

To add SiteMinder Support on 64-bit AIX

In this procedure, you replace the default 32-bit Webagent API libraries included with CA Identity Manager with the appropriate 64-bit libraries from the support site.

Note: The Webagent API libraries are four .so files distributed withCA Identity Manager (libsmagentapi.so, libsmcommonutil.so, libsmerrlog.so, and libsmjavaagentapi.so) and exist in the following location:
WebSphereRoot/IBM/WebSphere/Appserver/profiles/*AppServerName*/installedApps/*CellName*/IdentityMinder.ear/library

1. Download IBM AIX 6.1 64-bit IM WebAgent APIs as follows:
    a. Go to the CA support site (support.ca.com).
    b. Expand the Download Center navigation item on the left-hand side of the page.
    c. Select the Products link.
    d. Locate the first drop down in the grayed out area in the middle of the page, just below All Products.
    e. Select Identity Manager External User - MULTI-PLATFORM.
    f. For Select A Release, choose 12.5.
    g. For Select a Gen Level, choose SP6.
    h. Enable the checkbox called Show me published solutions for this release.
    i. Select CA Identity Manager from the selection box below the checkbox.
    j. Click Go.

2. On the resulting page look for the following:
   `IBM AIX 6.1 64-bit IM WebAgent API-ESD Only (GEN02170809E)`
3. Select Download on the right-hand side.

   On the resulting page, select a download method and download the ZIP file.
4. Expand the download ZIP file. You should see four WebAgent Library files:
   - libsmagentapi.so
   - libsmcommonutil.so
   - libsmerrlog.so
   - libsmjavaagentapi.so

Export and Import the EAR on 64-Bit AIX Systems

This procedure applies if you are integrating SiteMinder with CA Identity Manager on a 64-bit AIX system.

Export and import the EAR on a 64-Bit AIX system

1. Navigate to *deployment_manager_dir/*bin.
2. Be sure the Deployment Manager is running.
3. Export the deployed CA Identity Manager application, as follows:
   `./wsadmin -f imExport.jacl path-to-exported-ear`

   where *path-to-exported-ear* is the full path, including the file name of the exported EAR file.
4. Include the unzipped 64-bit Webagent Libraries (see page 51) as follows:
   a. Use unpackage.sh under WAS_HOME/WebSphere-tools to extract the exported ear file, the file from step 3.
   b. Copy the unzipped 64-bit Webagent Libraries to this location:
      `path-to-exported-ear/IdentityMinder.ear/library`
   c. Copy the updated IdentityMinder.ear folder to *WAS_HOME/WebSphere-ear* folder.
   d. Open console window and change the directory to *WAS_HOME*/WebSphere-tools
   e. Run package.sh
   f. Collect the repackaged EAR from the current location.
5. Copy the ims6Upgrade.jacl script from *WAS_HOME/*WebSphere-tools to the *deployment_manager_dir/*bin directory where:

   WAS_HOME is the directory where WebSphere is installed.

   deployment_manager_dir is the location where the Deployment Manager is installed.
6. On the Deployment Manager, deploy the updated IdentityMinder EAR as follows:
   a. From the command line, navigate to the following location:

      *deployment_manager_dir/*bin
   b. Be sure that the Deployment Manager is running.
   c. Run the ims6Upgrade.jacl script, as follows:
      `./wsadmin -f ims6Upgrade.jacl path-to-copied-ear cluster_name`

      Note: The ims6Upgrade.jacl script can take several minutes to execute.

Add Cluster Members

You can now add members to the cluster using the first cluster member as a template.

To add cluster members

1. In the Administrative Console for the Deployment Manager, go to Servers, Clusters.
2. Add a cluster member, selecting one of the nodes for which you created a profile.
3. Repeat this procedure for each cluster member you need to add to the cluster.

Configure Messaging Engines

On WebSphere 6.1, you configure a messaging engine on each cluster member.

To configure messaging engines

1. Start the Deployment Manager, which was stopped by the installation.
2. From the Deployment Manager, navigate to *was_home*/profiles/*Deployment_Manager_Profile*/bin.
3. Execute wsadmin as follows:

   `wsadmin -f ims6SetupClusterMember.jacl` *`NodeName ServerName ClusterName MEDataSourceJndiName`*

   MEDataSourceJndiName is a JNDI name that you want to assign for the messaging engine. A messaging engine is created for this server based on this name. For this name, use the format: *NodeName-ClusterMemberName*

   For example, using the sample field names used in this chapter, one JNDI name would be: was61013dmanNode01-was61013dman
4. Verify that the script completes with a "Save the Configuration" message and no errors.
5. Repeat steps 3 and 4 for each cluster member.
6. After you create the message engines, you create a message store (see page 55) for each messaging engine.

Create Message Stores

The Service Integration Bus includes a messaging engine for each cluster member. It manages the communications for the bus. Each messaging engine has a message store, for its exclusive use, where it records messages, subscription information, and transaction states.

Important! Configure a message store for each cluster member with one database per cluster member. The message store can be any database that WebSphere can access remotely, but it must be a XA provider. You can use the XA provider distributed with CA Identity Manager. If you use a different XA provider, see the WebSphere documentation for configuration details. A new schema must be created for each database and the database must be empty.

To create message stores

1. For this task you will need the following information:
   - The database name
   - The schema name
   - The username and password for the database runtime user
   - The username and password for the user who will execute the script with the DDL statements that create the schema
2. Use IBM's sibDDLGenerator command to generate the DDL statements needed to create the database resources used by the messaging engine.

   For instructions on using this utility, see IBM's documentation for the sibDDLGenerator.
3. After you create the DDL statements, you run a SQL script to import the schema into the database.
4. To enable the messaging engine to use the database, you configure an XA JDBC provider for the database type at the cluster scope level.
5. Configure a data source for the cluster in the WebSphere Console.

   Note the JNDI name when you create the data source.
6. In the Administrative Console for the Deployment Manager, go to System Administration, Nodes.
7. Verify that the nodes show a Synchronized status.
8. Test the connection to the data source, the node where you created the data source.
9. Repeat this procedure for each cluster member.

Note: After you create the message store, create core group policies (see page 56) which determine the distribution of messaging engines among servers in the cluster.

Create Core Group Policies

To enable high availability and workload management in the cluster, each messaging engine needs a core group policy. These policies control the distribution of the messaging engines, defining the preferred cluster member to use. If that cluster member fails, the messaging engine switches to another cluster member, but returns to the preferred cluster member when it becomes available.

Perform the following procedure once for each cluster member. The procedure provides the required steps to create a core group policy. For more information on this topic, see Setting up Preferred Servers in the Default Messaging Provider section of the WebSphere documentation.

To create a core group policy

1. In the WebSphere Console, locate the messaging engine for a cluster member.

   Make a note of the messaging engine information.

2. Navigate to the Default Core Group.
3. Create a new policy for the DefaultCoreGroup with the following settings:
   - Policy Type: One of N
   - Options: FailBack and Preferred Servers Only

   Note: Do not delete or modify the default policies.

4. Create a new match criteria for the policy you created with the following properties:
   - Name: type
   - Value: WSAF_SIB

5. Create another match criteria with the following properties.
   - Name: WSAF_SIB_MESSAGING_ENGINE
   - Value: *name of the messaging engine on the IMSBus*

     WebSphere automatically generates the name of the messaging engine when you create it. The name has the following format:

     cluster_name.00*n*-IMSBus

     where *cluster_name* is the name of the cluster you are configuring, and *n* represents a unique number for the messaging engine, which is automatically incremented each time a messaging engine is created for the cluster.

     For example, if the cluster name is im_cluster, and there are two messaging engines, the names would be:

     im_cluster.001-IMSBus

     im_cluster.002-IMSBus
6. Confirm that the message engine is assigned correctly:
   a. In the WebSphere console, locate the IMSBus in the service integration area.
   b. Select a message engine, then a message store.

      The message engine belongs to the cluster member when the JNDI name contains the cluster member's node name.
7. Return to the configuration page for the policy you are creating.
8. Select the cluster member you want to configure as the preferred cluster member for the new policy.

   You can select as many cluster members as needed from the cluster where the messaging engine is defined. Do not select node agents or the Deployment Manager.

   The first cluster member in the list is the one that the messaging engine will use by default. Move the cluster member up or down in the list until they appear in the order in which they should be used.
9. Click OK to save the changes.
10. Repeat this procedure for each cluster member.

Configure Workflow for Cluster Members

From the Deployment Manager system where you installed CA Identity Manager, you configure workflow for each cluster member.

To configure workflow for cluster members

1. Start the WebSphere Console.
2. Navigate to Application Servers, *server_name*, Communications, Expand Ports.
3. Edit Workpoint-client.properties file under IdentityMinder.ear/config.
4. Change the default port 2809 in the WebSphere section to the profile's port for the BOOTSRAP_ADDRESS.
5. Repeat this procedure for each cluster member.
6. Restart the cluster members.

Update the WebSphere Path for SiteMinder

Update the WebSphere Path definition for each cluster member if CA Identity Manager is integrated with Siteminder.

To update the WebSphere path

1. In the Deployment Manager, go to Application servers, *cluster_member*, Server Infrastructure, Java and process definition, Process Definition, Environment Entries.
2. Add the full path to the IdentityMinder.ear/library directory. For example, on UNIX, the path would be:
   `LIBPATH=`*`WebSphereRoot`*`/IBM/WebSphere/Appserver/profiles/`*`AppServerName`*`/installedApps/`*`CellName`*`/IdentityMinder.ear/library`
3. Repeat Steps 1 and 2 for each cluster member.

Configure the Proxy Plug-In for the Web Server

You install the proxy plug-in so that WebSphere can communicate with the web server.

To configure the proxy plug-in for the web server

1. See the WebSphere documentation for instructions about installing the proxy plug-in for the web server.

2. Restart the Web server to activate the plug-in.

   - For IIS Web Servers—In the master WWW service, be sure that the WebSphere plug-in (sePlugin) appears after the SiteMinder Web Agent plug-in and that the WebSphere plug-in started successfully.
   - For Sun Java System Web Servers—Be sure that the WebSphere plug-in (libns41_http.so) is loaded after the SiteMinder Web Agent plug-in (NSAPIWebAgent.so)

     For Sun Java System 6.0 Web Servers, check the order of plug-ins in <*sun_java_home*>/https-*instance*/config/magnus.conf.

     For Sun Java System 5.x Web Servers, copy the following lines from <*iplanet_home*>/https-*instance*/config/magnus.conf to <*iplanet_home*>/https-*instance*/config/obj.conf

     ```
     Init fn="load-modules" funcs="as_init,as_handler,as_term"
     shlib="/export/WebSphere/AppServer/bin/libns41_http.so"
     Init fn="as_init"
     bootstrap.properties="/export/WebSphere/AppServer/config/cells/plugin-cfg
     .xml"
     ```

     Add the following after AuthTrans fn="SiteMinderAgent" in the obj.conf file:

     ```
     Service fn="as_handler"
     ```
   - For Apache Web Servers— In the Dynamic Shared Object (DSO) Support section of *Apache_home*/config/httpd.conf, be sure that the SiteMinder Web Agent plug-in (mod2_sm.so)is loaded before the WebSphere plug-in (mod_ibm_app_server_http.so).

Start the WebSphere Cluster

To start the WebSphere cluster, you start the Deployment Manager and then start each managed node.

To start the WebSphere cluster

1. Start a Policy Server that supports CA Identity Manager.

   Note: If you have a Policy Server cluster, only one Policy Server should be running while you create Identity Manager directories, create or modify Identity Manager environments, or change WorkPoint settings.

2. Run the Deployment Manager.

3. On the first managed node, complete the following steps:

   a. Navigate to *was_home*\WebSphere\AppServer\bin.

   b. Execute the startNode.bat\.sh command.

      The first managed node starts.

4. Repeat step 2 on each node in the cluster.
5. Start each cluster member in Servers, Clusters, *cluster_name*, Cluster Members in the WebSphere Administrative Console on the Deployment Manager.
6. Make sure that the messaging engine for the cluster is running in Service integration, Buses, IMSBus, Messaging Engines in the WebSphere Admin Console on the Deployment Manager.
7. If you have installed a SiteMinder Web Agent, start the Web Server where you installed the SiteMinder Web Agent and the application server proxy plug-in.

Verify the Clustered Installation

When you have completed all steps and started the cluster, check that the installation was successful.

To verify the clustered installation

1. Access the Identity Manager Management Console as follows:

   `http://host_name:port/idmmanage`

   host_name

   Defines the fully-qualified host name for the server where CA Identity Manager is installed

   port

   Defines the application server port.

2. If these steps succeeded, start any extra Policy Servers and CA Identity Manager nodes that you stopped.

Configure a Remote Provisioning Manager

If you installed the Provisioning Manager on a different system from the Provisioning Server, you need to configure communication to the server.

Note: To install the Provisioning Manager, install the Identity Manager Administrative Tools on a Windows system.

To configure a remote Provisioning Manager

1. Log into the Windows system where you installed Provisioning Manager.
2. Go to Start, Programs, CA, Identity Manager, Provisioning Manager Setup.
3. Enter the hostname of the Provisioning Server.

4. Click Configure.
5. For an alternate Provisioning Server, select the domain name from the pull-down list.
6. Click Ok.

   You can now start the Provisioning Manager and see the domain name that you configured.

Install Optional Provisioning Components

Optional Provisioning Components for CA Identity Manager are in the im-pc-*release*.zip

release represents the current release of CA Identity Manager.

The ZIP files includes the following:

SPML Manager

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

SPML Service

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

Remote Agents

Run the specific agent installer from the Provisioning Component media (under \RemoteAgent) to install these components. If you want IPv6 support, you will need to install your agents.

Password Sync Agents

Run the Password Sync Agent installer from the Provisioning Component media (under \Agent) to install this component.

GINA

Run the GINA installer from the Provisioning Component media (under \Agent) to install this component.

Vista Credential Provider

Run the Vista Credential Provider installer from the Provisioning Component media (under \Agent) to install this component.

Bulk Loader Client/PeopleSoft Feed

Run the Bulk Loader Client installer from the Provisioning Component media (under \Clients) to install this component.

JCS SDK

Run the JCS SDK installer from the CA Identity Manager media (under \Provisioning) to install this component.

CCI Standalone

Run the CCI Standalone installer from the Provisioning Component media (under \Infrastructure) to install this component.

More information exists for these components in the following guides:

- Credential Provider (*Administration Guide)*
- GINA Option for Password Reset/Unlock (*Administration Guide)*
- Password Synchronization Agent (*Administration Guide)*
- Connector Xpress (*Connector Xpress Guide)*
- SPML Service (*Provisioning Reference Guide)*
- Agents for use with connectors (*Connectors Guide)*

Connector Xpress

To create your own connectors, you use Connector Xpress to create connectors without expertise required to use a programming interface.

Connector Xpress is a CA Identity Manager utility for managing dynamic connectors, mapping dynamic connectors to endpoints, and establishing routing rules for endpoints. You can use it to configure dynamic connectors to allow provisioning and management of SQL databases and LDAP directories.

Note: For more information on using Connector Xpress, see the *Connector Xpress Guide.*

Connectors

The Identity Manager installer installs all connectors by default. However, in some cases, you must install an agent on an endpoint system you are managing before you can use the related connector.

Connectors run on the Provisioning Server and communicate with the systems managed by an endpoint. For example, systems running Active Directory Services (ADS) can be managed only if the ADS Connector is installed on the Provisioning Server.

Note: For more information about each connector, see the *Connectors Guide*.

Chapter 5: Separate Database Configuration

This section contains the following topics:

Installation Status

This table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Basic installation ■ Installation on an application server cluster
X	**3. (Optional) Create separate databases.**
	4. (Optional) Install the Report Server.
	5. (Optional) Protect CA Identity Manager with SiteMinder.
	6. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

Create Separate Databases

CA Identity Manager requires a relational database to store objects and data for auditing, snapshots (reporting), workflow, and task persistence. When installing CA Identity Manager, all of the database schemas are created automatically when the application server is started. However, for scalability purposes, you may want to create a separate database to replace any one of the existing database schemas initially created by CA Identity Manager during installation.

You can create a new database instance for the following:

- Workflow
- Auditing
- Task Persistence
- Object Store
- Snapshots (reporting)
- Archive (task persistence archive)

Important! The Windows default locations for CA Identity Manager database schema files are the following:

- Workflow: run the CreateDatabase script (see page 70)
- Auditing: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db
- Task Persistence: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db
- Object Store: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db
- Snapshots (reporting): C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\imrexport\tools\db
- Archive (task persistence archive): C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db

How to Create Separate Databases

To create separate databases for CA Identity Manager:

✓	Step
	1. Create a MS SQL Server or Oracle database instance for CA Identity Manager.

✓	Step
	2. Edit the data source.
	3. (Optional) Run the SQL scripts.

Create an MS SQL Server Database Instance

To create an MS SQL Server Database Instance

1. Create a database instance in SQL server.
2. Create a user and grant this user the necessary rights (such as public and db_owner rights) to the database by editing the properties of the user.

   Note: The user must have at least select, insert, update, and delete permissions for all of the tables created by the .sql script for creating the database, and must be able to execute all of the stored procedures (if applicable) defined in these scripts. For example, the user must have these permissions on the tables defined in the following default location:

   C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\taskpersistence\sqlserver\idm_db_sqlserver.sql
3. While editing the user's properties, set the database you just created as the default database for the user.
4. Ensure the Authentication setting has a value of SQL Server on the Security tab of the SQL Server Properties dialog for the server where the database is installed.

Note: For complete information about MS SQL Server, see your MS SQL Server documentation.

Create an Oracle Database Instance

To create an Oracle Database Instance

1. Create a new tablespace.
2. Create a new user.

3. Grant the user rights to the new database.
   - Create/alter/drop tables
   - Create/alter/drop view
   - Create/alter/drop INDEX
   - Create/replace/drop stored procedures
   - Create/replace/drop functions
   - Create/drop sequence
   - Create/replace/drop triggers
   - Create/replace/drop types
   - Insert/select/delete records
   - CREATE SESSION / connect to database
4. Give DBA rights to the user.

Note: For complete information about Oracle, see your Oracle documentation.

Edit the Data Source

Important!When using WebSphere with Microsoft SQL Server, enable XA transactions. CA Identity Manager needs an XA data source for the database transactions to work properly. For more information on enabling XA transactions on Microsoft SQL Server, go to http://msdn.microsoft.com/en-us/library/aa342335.aspx http://msdn.microsoft.com/en-us/library/aa342335.aspx. Be sure to use JDBC driver version 1.2 compatible DLL files when enabling XA transactions.

To edit the data source

1. Within the WebSphere Administrative Console, open the appropriate data source descriptor.

   The JNDI names for the data source descriptors are as follows:
   - Task Persistence: jdbc/idm
   - Workflow: jdbc/WPDS
   - Auditing: auditDbDataSource
   - Snapshots: jdbc/reportsnapshot
   - Object Store: jdbc/objectstore
   - Archive: jdbc/archive

2. Change the DatabaseName, User, and Password in the data source descriptor to the appropriate values for the new database.

   The database schema (SQL scripts) are automatically applied when you restart CA Identity Manager.

3. Depending on your database, add the following to Custom Properties:

   - **SQL:** user=<username>, password=<password>, enable2Phase=true, selectMethod=cursor
   - Oracle: user=<username>, password=<password>

     Note: Ensure that the JDBC provider is created as XA.

The database schema (SQL scripts) are automatically applied when you restart CA Identity Manager or you can run the scripts to apply the changes now.

Run the SQL Scripts

SQL scripts are automatically run against the databases when CA Identity Manager starts, however if you want to run the SQL scripts yourself, perform the following steps before restarting the application server:

These scripts are installed with the Identity Manager Administrative Tools.

To run the SQL scripts

1. Do one of the following:
   - MS SQL Server: Open the Query Analyzer tool and select the script you need.
   - Oracle: Open the SQL prompt for the script you need.
2. Select one of the following scripts (shown with the default Windows locations) depending on what the database was created for:
   - Task Persistence:
     - MS SQL: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\taskpersistence\sqlserver\idm_db_sqlserver.sql
     - Oracle on Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\taskpersistence\oracle9i\idm_db_oracle.sql
     - Oracle on UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/db/taskpersistence/oracle9i/idm_db_oracle.sql

- Auditing:
  - MS SQL: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\auditing\sqlserver\ims_mssql_logs.sql
  - Oracle on Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\auditing\oracle\ims_oracle_logs.sql
  - Oracle on UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/db/auditing/oracle/idm_db_oracle.sql
- Snapshots:
  - MS SQL: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\sqlserver\ims_mssql_report.sql
  - Oracle on Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\oracle\ims_oracle_report.sql
  - Oracle on UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/imrexport/db/oracle/idm_db_oracle.sql
- Workflow: Run the SQL Scripts for Workflow (see page 70).

3. Run the script file.
4. Verify that no errors appeared when you ran the script.

Run the Script for Workflow

CA Identity Manager includes SQL scripts for setting up a new workflow database instance.

To run the CreateDatabase script

1. Add the path to the sqljdbc.jar to the DB_CLASSPATH attribute in the CreateDatabase.bat or .sh script before you run it.
2. From a command prompt, run CreateDatabase.bat or sh. The default location for this script is:

   Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\Workpoint\install.

   UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/Workpoint/install.

   A command prompt window and the WorkPoint application open.
3. Select the database type from the drop-down.

4. Use the following guidelines to fill in fields in the configuration utility:
   - For the JDBC Class parameter, enter:

     Oracle: oracle.jdbc.driver.OracleDriver

     SQL Server: com.microsoft.sqlserver.jdbc.SQLServerDriver
   - For the JDBC URL, enter:

     Oracle: jdbc:oracle:thin:@*wf_db_system*:1521:*wf_oracle_SID*

     SQL Server: jdbc:sqlserver://*wf_db_system*:1433; databaseName=*wf_db_name*
   - For the Database User ID parameter, enter the workflow user you created when creating the workflow database.
   - For the Password parameter, enter the password you created for the workflow user.
   - For the Database ID, enter WPDS
5. Accept the default check box selections.
6. Click the Initialize button.

   When the configuration is complete, a message that resembles the following appears in the Command Prompt window:

   ```
   The create database process finished with 0 errors.
   ```
7. Restart the application server.

Chapter 6: Report Server Installation

This section contains the following topics:

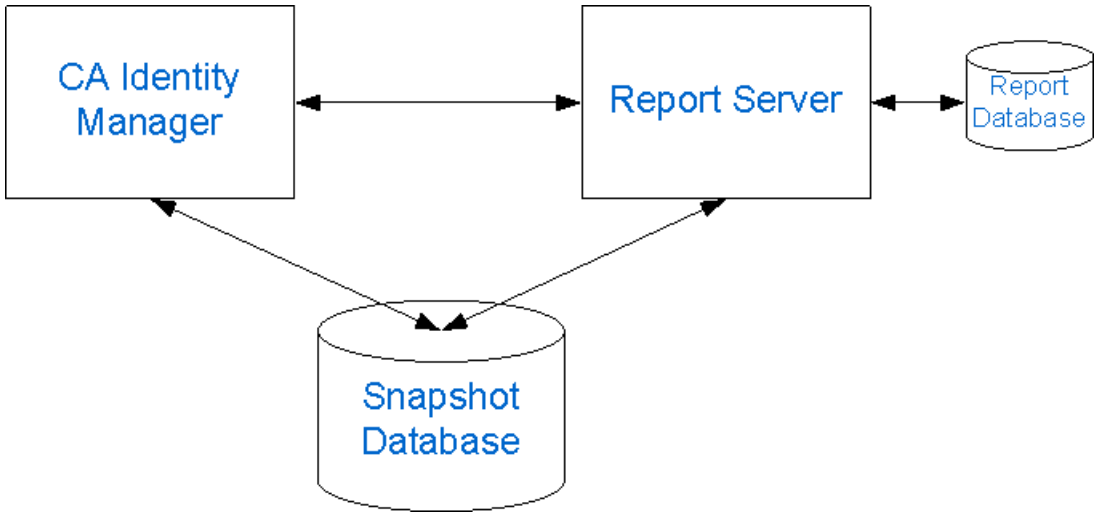
Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Basic installation ■ Installation on an application server cluster
	3. (Optional) Create separate databases.
X	**4. (Optional) Install the Report Server.**
	5. (Optional) Protect CA Identity Manager with SiteMinder.
	6. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

Reporting Architecture

In CA Identity Manager, the reporting setup requires the three major components in the following diagram:



Note: The Snapshot Database in this illustration graphic could also be the Audit Database or Workflow Database.

Report Server

Also known as CA Business Intelligence, this server generates reports, communicating directly with CA Identity Manager and the Snapshot Database.

Report Database

The database where the CA Report Server (Business Objects) stores its own data.

CA Identity Manager

CA Identity Manager allows you to export CA Identity Manager object data to the Report Database.

Snapshot Database

A separate database containing the snapshot data of objects in CA Identity Manager

Important! The Report Server is powered by Business Objects Enterprise. If you already have a Report Server in your environment and want to use it with CA Identity Manager, the minimum version required by CA Identity Manager is CA Business Intelligence 3.2.

Reporting Considerations

Consider the following before installing the Report Server:

- Installing the Report Server can take up to two hours.

- If JBoss is installed on the machine to which you are installing the Report Server, port conflicts may occur. If you experience port conflicts after installing the Report Server, you can locate JBoss port information in the following files:
  - jboss-service.xml

    Default location: *jboss_home*\server*server_configuration*\conf
  - server.xml

    Default location: *jboss_home*\server*server_configuration*\deploy\jboss-web.deployer

  jboss_home

  Specifies the JBoss installation path.

  server_configuration

  Specifies the name of your server configuration.

  Default value: default

  Note: Restart JBoss if you make changes to either of these files.

Hardware Requirements

The hardware requirements for the Report Server are based on the operating system:

- For UNIX, see the "Minimum Hardware Requirements" section in *installer-media-root-directory*/Docs/supported-platforms-SP3-Windows.pdf.
- For Windows, see the "Minimum Hardware Requirements" section in *installer-media-root-directory*/Docs/supported-platforms-SP3-Solaris.pdf.

Note: For more information about supported OS versions and databases, see the Business Objects web site.

How to Install the Report Server

The following checklist describes the steps to install CA Identity Manager's reporting feature:

✓	Step
	1. Review the report pre-installation checklist.
	2. Gather reporting information.
	3. Open ports required by the Report Server.

✓	Step
	4. Install the Report Server (CA Business Intelligence)
	5. Run the Registry Script.
	6. Copy the JDBC JAR files.
	7. Deploy the default reports.

Note: For more information on configuring reporting after the installation, see the *Administration Guide*.

Reports Pre-Installation Checklist

Print the following checklist to be sure that you meet the minimum system and database requirements before installing the Report Server:

- Be sure that the Windows or UNIX system on which you are installing the Report Server meets the minimum system requirements.
- Be sure that you use MySQL for the Report Database, which is bundled with the Report Server Installer.
- If you create a database instance for the Snapshot Database, run the following scripts on the new database:
  - Microsoft SQL: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\sqlserver\ims_mssql_report.sql
  - Oracle: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\oracle\ims_oracle_report.sql

    To execute these scripts, the database user needs DBA, connect, and resource roles and system privileges to create tables, indexes, sessions and views with global query rewrite privilege.
- On UNIX, set the following parameters as global in the local .profile files:
  - ORACLE_BASE: the top-level directory where Oracle is installed.
  - ORACLE_HOME: the path to the Oracle root directory under ORACLE_BASE
  - LD_LIBRARY_PATH: $ORACLE_HOME/lib32:$ORACLE_HOME/lib

    If Oracle is a 64-bit installation, use lib32. Use SQL Plus to connect to the oracle database instance to check if it is a 64-bit installation.

- ORACLE_SID: the SID name used in the tnsnames.ora file.
- JAVA_HOME: the path to the Java root directory. Business Objects installs a JDK in the following location:

  `report_server_home/jre`

  Note: JDK 1.5 is the minimum version supported for reports.
- PATH: $LD_LIBRARY_PATH:$JAVA_HOME:$JAVA_HOME/bin:$ORACLE_HOME/bin:$PATH
- LC_ALL: en_US.UTF-8

  Note: Be sure that the CASHCOMP environment variable is empty.

- On UNIX systems:
  - 3 GB of free space is required under /tmp.
  - You need access to a non-root user account to install the Report Server.

    This user should have a home directory in the local file system. For example. the following command creates a user with a local home directory:

    `useradd -u 505 -g 0 -d /export/home/cabi -m cabi`

    Also, add the non-root user to the oinstall group and any group for which the root user is a member.
  - Enter the database server name in the /etc/hosts file if the database server is not on the same system as the Report Server.
  - If you encounter problems, check the SDK.log under these locations:

    /opt/CA/SharedComponents/CommonReporting3/ca-install.log

    /opt/CA/SharedComponents/CommonReporting3/CA_Business_Intelligence_InstallLog.log

Reporting Information

Record the following information you need during the Report Server installation:

Field Name	Description	Your Response
Administrator Password	Defines the password to log in to the Business Objects Infoview console.	
User Name	Identify the username for the Report Database.	

Field Name	Description	Your Response
Password	Identify the administrative password credentials for the Report Database.	
Pre-Installed Tomcat Information	Identify the path and port numbers for any previous installation of Tomcat. If you do not want to use a previous installation of Tomcat, Report Server installer can install Tomcat.	
Tomcat Port Numbers	The Tomcat connection, redirect, and shutdown ports. **Note:** If you install the Report Server on the same system as the CA Identity Manager, be sure that the Tomcat connection port does not conflict with the port number you specified for the application server URL when installing the CA Identity Manager.	

Open Ports for the Report Server

For CA Identity Manager and the Report Server to communicate successfully, the following ports must be opened.

- The Central Management Server (CMS) port: 6400
- The Report Server web application port:
  - JBoss/Tomcat: 8080
  - WebLogic: 7001
  - WebSphere: 9080

Note the following:

- This port is not the application server port for the Identity Manager Server.
- The web server ports are provided during the Report Server installation. If you use different ports during the installation, those ports must be opened through the firewall when the Report Server is deployed in production.
- The Report Server does not connect to the application server used by CA Identity Manager.

- All database ports that CA Identity Manager has configured for the reporting and auditing databases. The Identity Manager Server must send database information to the Report Server, so these ports must be opened. For example, if the Snapshot Database is an Oracle database, the Report Server needs the Oracle port open outbound.

Install the CA Report Server

You can install the Report Server on a supported Windows or UNIX system. The following sections detail how to install the Report Server using a Windows and UNIX installation wizard.

Important! For a production environment, install the Report Server on a separate system from the system with the Identity Manager Server. If you want to install the Report Server on the same system as the Identity Manager Server for demonstration purposes, choose non-default ports for 8080 and 1099.

The Report Server is powered by Business Objects.

Note: CA Identity Manager supports the latest version of Business Objects XI. For more information on upgrading the Report Server, see the *Upgrade Guide*.

Run the Windows Installer

Install the Report Server using the Windows installation wizard (Disk1\InstData\VM\Install.exe) found on the Report Server media.

Note: The Report Server is available for download on the CA Support site, under CA Identity Manager product downloads.

To install the Report Server

1. Exit all applications.
2. Download the Report Server and unzip it.
3. Navigate to Disk1\InstData\VM and double-click the installation executable.

   The installation wizard starts.

4. Use the gathered reporting information to install the Report Server.

   Note the following:

   - Select a New install during installation. This ensures that you use MySQL as the Report Database. If you need to set non-default ports to avoid port conflicts, select a Custom install, but be sure to select MySQL for the Report Database.
   - Choose Tomcat as the web server, and deselect IIS.
   - If you are installing the Report Server on the same system as CA Identity Manager, be sure that the Tomcat connection port does not conflict with the port number you specified for the application server URL when installing CA Identity Manager. However, we recommend installing the Report Server on a different system than the Identity Manager Server in a production environment.

5. Review the installation settings and click Install.

   The Report Server is installed.

Run the UNIX Installer

Add execute permissions to the install file by running the following command:

```
chmod+x /ca-iamreportserver-12.5-solaris/cabiinstall.sh
```

Important! The installer may crash if executed across different subnets. To avoid this problem, install the Report Server directly on the host computer.

To install the Report Server

1. Log in as the non root user you created to install the Report Server.
2. Exit all applications.
3. Download the Report Server and untar it.

   Note: The Report Server is available for download on the CA Support site, under CA Identity Manager product downloads.

4. Open a command window and navigate to where the install program is located.
5. Enter the following commands:

   ```
   /ca-iamreportserver-12.5-solaris/cabiinstall.sh gui
   ```

6. Use the gathered reporting information to install the Report Server.

   Note the following:

   - Select a New install during installation. This ensures that you use MySQL as the Report Database. If you need to set non default ports to avoid port conflicts, select a Custom install, but be sure to select MySQL for the Report Database.
   - Choose Tomcat as the web server.
   - If you are installing the Report Server on the same system as CA Identity Manager, ensure that the Tomcat connection port does not conflict with the port number you specified for the application server URL when installing CA Identity Manager.
   - The installer installs the Report Server to /opt/CA/SharedComponents/CommonReporting3. Specifying another location does not change the installation location. So the /opt/CA directory must have non-root user permissions or the installation fails.

7. Review the installation settings and click Install.

   The Report Server is installed.

8. Click Done.

Run the Registry Script

For CA Identity Manager to change data sources for reports in the Report Server, run the mergeConnection script.

Note: On a 64-bit system, omit this procedure. The Report Server is a 32-bit application, so you use the 32-bit side of the registry. Open REGEDT32 directly from SysWOW64, and create the MergeConnectionProperties key with the Type REG_SZ and value Yes. Create the key in this location:

```
@HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Business Objects\Suite 12.0\Crystal Reports\DatabaseOptions
```

On the Report Server, the default location for this script is as follows:

- Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\ReportServerTools.
- UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/ReportServerTools.

On Windows, perform the following steps:

1. Run the mergeconnections_3.0.reg script and respond to the prompts that appear.
2. Click Start, Program Files, CA, Report Server, Central Configuration Manager.
3. Start all services, including Tomcat and the BO Server service.

On UNIX, perform the following steps:

1. Check for Windows control characters in the mergeconnections_3.0.cf script.

   If you downloaded the software using FTP in binary mode, these characters should not exist in this script. If you used another download method, use the dos2unix command to remove these characters.
2. Copy the mergeconnections_3.0.cf script from the ReportServerTools directory to the following directory

   installation-directory/bobje/enterprise120/generic
3. Source in the environment variables for BusinessObjects Enterprise, as follows:

   source *installation-directory*/bobje/setup/env.sh
4. Run the following script, as follows:

   ./configpatch.sh mergeconnections_3.0.cf

   Select 1 as the option when prompted.
5. Restart crystal processing servers as follows:
   a. Log in as the non root user you used to install the Report Server.
   b. Issue these commands:

      ```
      cd /opt/CA/SharedComponents/CommonReporting3/bobje
      ./stopservers
      ./startservers
      ```

Copy the JDBC JAR Files

To copy the JDBC JAR files

1. Navigate to the jdbcdrivers folder on the CA Identity Manager media, as follows:
   - Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\lib\jdbcdrivers
   - UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/lib/jdbcdrivers
2. Copy ojdbc14.jar (for Oracle) or sqljdbc.jar (for SQL Server) to the following location:
   - Windows: C:\Program Files\CA\SC\CommonReporting3\common\4.0\java\lib
   - UNIX: /opt/CA/SharedComponents/CommonReporting3/bobje/java/lib

3. Open the CRConfig.xml file in the following location:
   - Windows: C:\Program Files\CA\SC\CommonReporting3\common\4.0\java
   - UNIX: /opt/CA/SharedComponents/CommonReporting3/bobje/java
4. Add the location of the JDBC JAR files to the Classpath. For example:
   - Windows: <Classpath>report_server_home\common\4.0\java\lib\sqljdbc.jar; report_server_home\ common\4.0\java\lib\ojdbc14.jar ...</Classpath>
   - UNIX: <Classpath>${BOBJEDIR}/java/lib/sqljdbc.jar:${BOBJEDIR}/java/lib/ojdbc14.jar: ...</Classpath>
5. Save the file.
6. Restart the Report Server as follows:
   - For Windows, do the following:
     a. Go to Start, Program Files, CA, Report Server, Central Configuration Manager.

        The Central Configuration Manager opens.

     b. Select all services and click Restart.
   - For UNIX, do the following:

```
cd /opt/CA/SharedComponents/CommonReporting3/bobje
./stopservers
./startservers
```

Deploy Default Reports

CA Identity Manager comes with default reports you can use for reporting. BIConfig is a utility that uses a specific XML format to install these default reports for CA Identity Manager.

If you are upgrading, first remove the CA Identity Manager reports folder using the Central Management Console. The default reports from the previous installation do not work. You can then deploy default reports for the new Report Server.

Important! This process will update all default reports. If you customized any default reports, be sure to back them up before performing the update.

To deploy the default reports

1. Gather the following information about the Report Server:
   - Hostname
   - Administrator name

- Administrator password
- Snapshot database type

2. Copy the contents of the biconfig folder from Reports *installer-root-directory*/disk1/cabi to *im_admin_tools_dir*/ReportServerTools.
3. Make sure that the JAVA_HOME variable is set to the 32-bit version of the JDK1.5 you installed.
4. Run one of the following commands:
   - For a Microsoft SQL Snapshot Database:

     ```
     biconfig -h "hostname" -u "administrator_name" -p "administrator_password"
     -f "ms-sql-biar.xml"
     ```

   - For an Oracle Snapshot Database:

     ```
     biconfig -h "hostname" -u "administrator_name" -p "administrator_password"
     -f "oracle-biar.xml"
     ```

   Note: In a UNIX operating environment, be sure that biconfig.sh has execute permissions.
5. View the biconfig.log file found in the location where you ran the command in Step 4.
6. Verify that the default reports installed successfully. Inspect the end of the log file for status; a successful install appears as follows:

   ```
   ReportingDeployUtility - Reporting utility program terminated and return code =
   0
   ```

Verify the Reporting Installation

To verify that reporting has been installed correctly, do the following:

- In the Central Management Console, be sure that all services are running.
- Be sure that your Report Database is running.

Note: For more information on configuring reporting after the installation, see the *Administration Guide*.

Silent Installation

For more information about silent installation of the Report Server, see the *CA Business Intelligence Installation Guide.* The Report Server documentation is available in one of the following locations when you extract the Report Server installer files:

- **Windows**: *install_root_directory*\Docs\ENU\CABI_Impl_ENU.pdf
- **UNIX**: *install_root_directory*/Docs/ENU/CABI_Impl_ENU.pdf

How to Uninstall Reporting

You uninstall the Report Server when it is no longer required on the system. See the *CA Business Intelligence* documentation.

After uninstalling the Report Server, remove leftover items (see page 85).

Remove Leftover Items

The following sections detail the items you must manually remove after uninstalling the Report Server to keep the system as clean as possible and to prevent a reinstallation of the Report Server to the same machine from failing.

Remove Windows Items

To remove leftover Report Server items on Windows

1. Navigate to *report_server_home*.

   report_server_home specifies the Report Server installation path.
2. Open the BusinessObjects Enterprise 12.0 folder, and delete the following folders:
   - Data
   - java
   - Logging
   - Samples
   - Web Content
   - Web Services
   - win32x86
3. Return to the Report Server folder.
4. Open the common folder.
5. Open the 4.0 folder, and delete the following folders:
   - crystalreportviewers12
   - java

You have completed removing leftover items.

Remove UNIX Items

To remove leftover Report Server items on UNIX

1. Navigate to the following location from a command prompt:

   /opt/CA/SharedComponents

2. Delete the CommonReporting3 folder.

You have completed removing leftover items.

Chapter 7: SiteMinder Configuration

This section contains the following topics:

Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Basic installation ■ Installation on an application server cluster
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
X	**5. (Optional) Protect CA Identity Manager with SiteMinder.**
	6. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

How Resources are Protected

Advanced authentication requires you to use a SiteMinder Policy Server in your implementation.

In many situations, the application server hosting the Identity Manager Server is on a separate system from the one with the Web Server that proxies requests to the application server. To provide forwarding services, the Web Server needs the following:

- A plug-in that is provided by the application server vendor
- A SiteMinder agent to protect the CA Identity Manager resources, such as the User Console, Self Registration, and the Forgotten Password feature

The Web Agent controls the access of users who request CA Identity Manager resources. After authenticating and authorizing users, the Web Agent allows the Web Server to process the requests.

When the Web Server receives the request, the application server plug-in forwards it to the application server hosting the Identity Manager Server.

The Web Agent facilitates communication between the Identity Manager Server and the Policy Server and protects CA Identity Manager resources that are exposed to users and administrators.

How to Protect CA Identity Manager with SiteMinder

The following table describes the steps involved in configuring SiteMinder to protect CA Identity Manager resources:

✓	Step
	1. Be sure you have installed the Identity Manager extensions on the SiteMinder Policy Server as described in the Installation Prerequisites chapter.
	2. Install a SiteMinder Web Agent to protect CA Identity Manager resources.
	3. Install the plug-in the Web Server uses to forward requests to the application server.
	4. Configure the SiteMinder Policy Store for use with CA Identity Manager.
	5. Start the application server and other servers in the installation.
	6. Verify that the plug-in is successfully forwarding requests to the application server.

✓	Step
	7. (Optional) Configure SiteMinder high availability for CA Identity Manager.

Install the SiteMinder Web Agent

You can use a SiteMinder Web Agent or a Web Agent Group to protect CA Identity Manager resources. For supported Web Agent versions, see the CA Identity Manager support matrix on the CA Support Site.

Note: For more information about Web Agent groups, see the *CA SiteMinder Web Access Manager Policy Server Configuration Guide*.

Before installing the Web Agent, ensure the following requirements have been met:

- The SiteMinder Policy Server is installed and configured.
- The system that hosts the Web Agent has network access to the Policy Server.
- The Web Server that hosts the Web Agent is running.

The following table lists the steps to install and configure a SiteMinder Web Agent:

✓	Step	Refer to...
	1. Install and configure the Web Agent.	*CA SiteMinder Web Access Manager Web Agent Installation Guide*
	2. If you installed the Web Agent on an IIS Web Server, be sure to set the DefaultAgentName and DefaultPassword parameters of your Agent Configuration Object.	*CA SiteMinder Web Access Manager Web Agent Installation Guide*
	3. Enable the Web Agent.	*CA SiteMinder Web Access Manager Web Agent Installation Guide*
	4. If you are using an IIS web server, ensure the SiteMinder web agent ISAPI filter appears before any other filter, including the SePlugin filter, in the IIS console.	IIS documentation

Important! CA Identity Manager now uses a new CA styles EAR. To support this, change the web server plug-in that is used to forward to the application server, by adding a redirection to /castylesr5.1.1 in addition to /idm in the http proxy forwarder.

To use the SiteMinder Web Agent to protect CA Identity Manager, select the Web Agent when you create an Environment. For instructions, see the *Configuration Guide*.

Note: You do not need to create any additional objects in SiteMinder to use the SiteMinder Web Agent.

To verify the Web Agent, confirm the following:

- The SiteMinder Policy Server Authentication and Authorization logs verify that the Web Agent starts properly.
- The Agent log for the Web Agent verifies that the Web Agent starts properly.

Install the Proxy Plug-In for WebSphere

Once the Web Agent authenticates and authorizes a request for a CA Identity Manager resource, the Web Server on which you installed the Web Agent must forward the request to the application server that hosts the Identity Manager Server. This is accomplished through a Web Server proxy plug-in provided by the application server vendor.

After you install the CA Identity Manager components and deploy the IdentityMinder EAR, you update the plug-in using WebSphere's GenPluginCfg command.

To install the proxy plug-in

1. Install the proxy plug-in from the WebSphere Launch Pad.
2. Add the Web Server to the WebSphere cell by running the configurewebserver1.bat command as follows:
   a. Edit *websphere_home*\Plugins\bin\configurewebserver1.bat/,sh in a text editor.
   b. Add a user name and password to after wsadmin.bat/.sh as follows:
      `wsadmin.bat -user wsadmin -password password -f configureWebserverDefinition.jacl`
   c. Run configurewebserver1.bat/.sh.

   Note: See the IBM WebSphere documentation for more information about the configurewebserver command.

3. From the command line, navigate to *websphere_home*\bin, where *websphere_home* is the installed location of WebSphere.

   For example:

   - **Windows:**

     C:\Program Files\WebSphere\AppServer\bin\

   - **UNIX:**

     /*home_dir*/WebSphere/AppServer/bin

4. Run the GenPluginCfg.bat or GenPluginCfg.sh command.

   Running this command generates a plugin-cfg.xml file in *websphere_home*\Plugins\config\webserver1\config\cells.

5. If the application server is on a separate system from the one with the Web server, copy the plugin-cfg.xml to the following directory on the system where you installed the proxy plug-in:

   websphere_home\AppServer\profiles*server_name*\config\cells*websphere_cell*\nodes\webserver1_node\servers\webserver1\

6. Restart the Web server to activate the plug-in as follows:

   - IIS Web Servers: In the master WWW service, ensure that the WebSphere plug-in (sePlugin) appears after the SiteMinder Web Agent plug-in and that the WebSphere plug-in started successfully.
   - iPlanet Web Servers: Ensure that the WebSphere plug-in (libns41_http.so) is loaded after the SiteMinder Web Agent plug-in (NSAPIWebAgent.so)

     For iPlanet 6.0 Web Servers, check the order of plug-ins in *iplanet_home*/https-*instance*/config/magnus.conf.

     For iPlanet 5.x Web Servers, copy the following lines from *iplanet_home*/https-*instance*/config/magnus.conf to *iplanet_home*/https-*instance*/config/obj.conf

     Init fn="load-modules" funcs="as_init,as_handler,as_term" shlib="/export/WebSphere/AppServer/bin/libns41_http.so"

     Init fn="as_init" bootstrap.properties="/export/WebSphere/AppServer/config/cells/plugin-cfg.xml"

     Add the following after AuthTrans fn="SiteMinderAgent" in the obj.conf file:

     Service fn="as_handler"

   - Apache Web Servers: In the Dynamic Shared Object (DSO) Support section of *apache_home*/config/httpd.conf, be sure that the SiteMinder Web Agent plug-in (mod2_sm.so) is loaded before the WebSphere plug-in (mod_ibm_app_server_http.so).

Configure the Policy Store for CA Identity Manager

Once you install the CA Identity Manager Extensions for SiteMinder on the system with the Policy Store, extend the policy store schema for CA Identity Manager.

To extend the schema to the policy store, use the Identity Manager Administrative Tools. Install the tools using the CA Identity Manager installation program, without installing the Identity Manager Server.

Configure a Relational Database

To configure a relational database policy store

1. Configure the directory as a supported SiteMinder Policy Store.

   Note: Be sure that SiteMinder is pointing to this policy store. For configuration instructions, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide.*

2. Run one of the following scripts for CA Identity Manager on the Policy Store database:

   - **SQL:** C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas\MicrosoftSQLServer\ims8_mssql_ps.sql
   - **Oracle:** /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/policystore-schemas/OracleRDBMS/ims8_oracle_ps.sql

   The preceding are default installation locations. The location for your installation may be different.

Configure Sun Java Systems Directory Server or IBM Directory Server

To configure a Sun Java Systems Directory or IBM Directory policy store

1. Configure the directory as a supported SiteMinder Policy Store.

   Note: Be sure that SiteMinder is pointing to this policy store. For configuration instructions, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide.*

2. Add the appropriate LDIF schema file from the following table to the directory. The default Windows location for the LDIF files is C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas.

   Adding the following schema files for your directory:

   - **IBM Directory Server:**

     IBMDirectoryServer\V3.identityminder8

   - **Sun Java Systems Directory Server (iPlanet):**

     SunJavaSystemDirectoryServer\sundirectory_ims8.ldif

Configure Microsoft Active Directory

To configure a Microsoft Active Directory policy store, you apply the activedirectory_ims8.ldif script.

To configure an Active Directory policy store

1. Configure the directory as a supported SiteMinder Policy Store.

   Note: Be sure that SiteMinder is pointing to this policy store. For configuration instructions, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide.*

2. Modify the activedirectory_ims8.ldif schema file as follows:

   a. In a text editor, open the activedirectory_ims8.ldif file. The default Windows location is:

      C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas\MicrosoftActiveDirectory

   b. Replace all instances of {root} with the root organization for the directory.

      The root organization must match the root organization that you specified when you configured the policy store in the Policy Server Management Console.

      For example, if the root is dc=myorg,dc=com, replace dn: CN=imdomainid6,CN=Schema,CN=Configuration,{root} with dn: CN=imdomainid6,CN=Schema,CN=Configuration,dc=myorg,dc=com

   c. Save the file.

3. Add the schema file as described in the documentation for your directory.

Configure Microsoft ADAM

To configure a Microsoft ADAM policy store, you apply the adam_ims8.ldif script.

To configure a Microsoft ADAM policy store

1. Configure the directory as a supported SiteMinder Policy Store.

   Note: Be sure that SiteMinder is pointing to this policy store. For configuration instructions, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide.*

2. Modify the adam_ims8.ldif schema file as follows:

   a. In a text editor, open the adam_ims8.ldif\.ldif file. The default Windows location is:

      C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas\MicrosoftActiveDirectory

b. Replace every cn={guid} reference with the string you found when you configured the SiteMinder policy store in Step 1 of this procedure.

For example, if the guid string is CN={39BC711D-7F27-4311-B6C0-68FDEE2917B8}, then replace every cn={guid} reference with CN={39BC711D-7F27-4311-B6C0-68FDEE2917B8}.

c. Save the file.

3. Add the schema file as described in the documentation for your directory.

Configure CA Directory Server

To configure a CA Directory policy store

1. Configure the directory as a supported SiteMinder Policy Store.

   Note: Be sure that SiteMinder is pointing to this policy store. For configuration instructions, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide.*

2. Copy etrust_ims8.dxc to *dxserver_home*\config\schema

   where *dxserver_home* is the directory where CA Directory is installed. The default source location for this file on Windows is C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas\eTrustDirectory.

3. Create a custom schema configuration file as follows:

   a. Copy the *dxserver_home*\config\schema\default.dxg to *dxserver_home*\config\schema*company_name*-schema.dxg.

   b. Edit the *dxserver_home*\config\schema*company_name*-schema.dxg file by adding the following lines to the bottom of the file:

   ```
   # Identity Manager Schema
   source "etrust_ims8.dxc";
   ```

4. Edit the *dxserver_home*\bin\schema.txt file by adding the contents of etrust_ims_schema.txt to the end of the file. The default source location for this file on Windows is C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas\eTrustDirectory.

5. Create a custom limits configuration file as follows:

   a. Copy the *dxserver_home*\config\limits\default.dxc to *dxserver_home*\config\limits*company_name*-limits.dxc.

   b. Increase the default size limit to 5000 in the *dxserver_home*\config\limits*company_name*-limits.dxc file as follows:

   ```
   set max-op-size=5000
   ```

   Note: If you upgrade CA Directory, the limits.dxc file is overwritten, therefore you must reset max-op-size to 5000 after the upgrade is completed.

6. Edit the *dxserver_home*\config\servers*dsa_name*.dxi as follows:

```
# schema
source "company_name-schema.dxg";

#service limits
source "company_name-limits.dxc";
```

   where *dsa_name* is the name of the DSA using the customized configuration files.

7. Run the dxsyntax command.

   This utility reports any errors with the directory configuration. If this utility runs with no errors, continue to Step 8.

8. Stop and restart the DSA as the dsa user to make the schema changes take effect, as follows:

```
dxserver stop dsa_name
dxserver start dsa_name
```

Configure Novell eDirectory Server

To configure an Novell eDirectory Server policy store, you apply the novell_ims8.ldif script.

To configure an Novell eDirectory policy store

1. Configure the directory as a supported SiteMinder Policy Store.

   Note: Be sure that SiteMinder is pointing to this policy store. For configuration instructions, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide.*

2. Find the DN of the NCPServer for your Novell eDirectory Server by entering the following information in a command window on the system where the Policy Server is installed:

```
ldapsearch -h hostname -p port -b container -s sub
-D admin_login -w password objectClass=ncpServer dn
```

   For example:

```
ldapsearch -h 192.168.1.47 -p 389 -b "o=nwqa47container" -s sub -D
"cn=admin,o=nwqa47container" -w password objectclass=ncpServer dn
```

3. Open the novell_ims8.ldif file.

4. Replace every NCPServer variable with the value you found in Step 2.

   The default location for novell_ims8.ldif on Windows is:

   C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas\NovelleDirectory

   For example, if the DN value is cn=servername,o=servercontainer, you would replace every instance of *NCPServer* with cn=servername,o=servercontainer.

5. Update the eDirectory Server with the novell_ims8.ldif file.

   See the Novell eDirectory documentation for instructions.

Configure Oracle Internet Directory (OID)

To configure an Oracle Internet Directory policy store

1. Configure the directory as a supported SiteMinder Policy Store.

   Note: Be sure that SiteMinder is pointing to this policy store. For configuration instructions, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide.*

2. Update the Oracle Internet Directory Server with the oracleoid_ims8.ldif file. The default installation location for this file on Window is:

   C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\policystore-schemas\OracleOID\

   See the Oracle Internet Directory documentation for instructions.

3. Start the Policy Server services as follows:

   a. Open the Policy Server Management Console.

   b. Click the Update button in the console and verify that the services started successfully.

Note: If you experience a timeout when searching for Admin roles using the wildcard (*) character, create a SearchTimeout string value in the LdapPolicy key in the registry. Set the value to a number greater than 20 seconds, which is the default search timeout, then restart the Policy Server services.

To access the registry on Windows, open Start, Run. Enter REGEDT32 in the Run window. On Solaris, open *policy_server_home*/registry/sm.registry.

The LdapPolicy key is located in:

HKEY_LOCAL_MACHINE\SOFTWARE\Netegrity\SiteMinder\CurrentVersion\Ds\

Verify the Policy Store

To verify the policy store, confirm the following:

- Your Policy Server log does not contain a section of warnings that begins with the following:

  ```
  *** IMS NO SCHEMA BEGIN
  ```

  Note: For SiteMinder r6.x, check smps.log.

  This warning appears only if you have installed the Extensions for the SiteMinder Policy Server, but you have not extended the Policy Store schema.

- The CA Identity Manager objects exist in the policy store database or directory. The CA Identity Manager objects begin with an ims prefix.

Start the Servers for WebSphere

You start the servers in your WebSphere implementation so that it is available for use.

To start the servers

1. If you installed a SiteMinder Policy Server, start the Policy Server that supports CA Identity Manager.

   Note: If you have a Policy Server cluster, only one Policy Server should be running while you create Identity Manager directories, create or modify Identity Manager environments, or change WorkPoint settings.

2. Run the Deployment Manager if you have a WebSphere cluster.

   If have only a single node installation, skip to Step 7.

3. On the first managed node, complete the following steps:

   a. Navigate to *was_home*\WebSphere\AppServer\bin.

   b. Execute the startNode.bat\.sh command.

      The first managed node starts.

4. Repeat Step 2 on each node in the cluster.

5. Start each cluster member in Servers, Clusters, *cluster name*, Cluster Members in the WebSphere Administrative Console on the Deployment Manager.

6. Be sure that the messaging engine for the cluster is running in Service integration, Buses, IMSBus, Messaging Engines in the WebSphere Admin Console on the Deployment Manager.

7. Start the Web Server where you installed the SiteMinder Web Agent and the application server proxy plug-in.

Verify SiteMinder Configuration

The Identity Manager Server installation contains a JSP page that you can use to verify that the application server connector is successfully forwarding requests to the application server.

In a browser, enter the following URL:

http://*web_server*/idm/ui/ping.jsp

For example:

http://MyServer.MyCompany.com/idm/ui/ping.jsp

If your application server connector is functioning, you receive a JSP page with an initial heading of Request Information. This page provides details about the processing of the request for the JSP page.

If the Web Agent you created is functioning correctly, information similar to the following appears under Request Headers in the page displayed in your browser:

```
SM_AUTHTYPE = Not Protected
SM_DOMAIN = domain
SMTRANSACTIONID = system-generated_id
```

For example:

```
SM_AUTHTYPE = Not Protected
SM_DOMAIN = .MyCompany.com
SMTRANSACTIONID = 41041aac-04ec-3edbc669-0a70-012d19d9
```

Configure SiteMinder High Availability for a WebSphere Cluster

If you have created a SiteMinder Policy Server cluster, you can configure the WebSphere cluster to use it for load balancing and failover.

To configure SiteMinder high availability for a WebSphere cluster

1. Edit the ra.xml file in this location:
   WAS_PROFILE/config/cells/*CELL_NAME*/applications/IdentityMinder.ear/deployments/IdentityMinder/policyserver_rar/META-INF
2. Modify these items, which are explained in the sections that follow:
   - Connection settings for the Policy Server
   - The number of Policy Servers
   - The selection of load balancing or failover for the cluster.
3. Repeat these steps for each Identity Manager server in the cluster.
4. Restart the WebSphere server for changes to take effect.

Modify Policy Server Connection Settings

The Policy Server connection information should reflect the primary server for the production environment. This information consists of the ConnectionURL, the user name and password for the SiteMinder Admin account, and the name and shared secret for the Agent.

In the following example, the values to edit appear in CAPITAL LETTERS.

```
<config-property>
        <config-property-name>ConnectionURL</config-property-name>
   <config-property-type>java.lang.String</config-property-type>
        <config-property-value>DEVELOPMENT.SEVERCOMPANY.COM,VALUE,VALUE,VALUE</co
nfig-
                property-value>
</config-property>

<config-property>
        <config-property-name>UserName</config-property-name>
   <config-property-type>java.lang.String</config-property-type>
        <config-property-value>SITEMINDER-ADMIN-NAME</config-property-
                value>
</config-property>
```

```
<config-property>
        <config-property-name>AdminSecret</config-property-name>
        <config-property-type>java.lang.String</config-property-type>
        <config-property-value>ENCRYPTED-PASSWORD</config-
                property-value>
</config-property>
<config-property>
        <config-property-name>AgentName</config-property-name>
        <config-property-type>java.lang.String</config-property-type>
        <config-property-value>DEVELOPMENT-AGENT-NAME</config-property-
                value>
</config-property>

<config-property>
        <config-property-name>AgentSecret</config-property-name>
        <config-property-type>java.lang.String</config-property-type>
        <config-property-value>ENCRYPTED-AGENT-SECRET</config-
                property-value>
</config-property>
```

Note: For the values that require encrypted text, use the Identity Manager password tool. For more information, see the *Configuration Guide*.

Add More Policy Servers

To add more Policy Servers to the CA Identity Manager installation instance, edit the FailoverServers entry in the ra.xml file.

Note: Include the primary Policy Server and all failover servers in the FailoverServers entry.

For each Policy Server, enter an IP address followed by port numbers for authentication, authorization, and accounting services. Use a semi-colon to separate entries as shown here:

```
<config-property>
        <config-property-name>FailoverServers</config-property-name>
        <config-property-type>java.lang.String</config-property-type>
        <config-property-value>
                172.123.123.123,44441,44442,44443;172.123.123.124,33331,
                33332,33333
        </config-property-value>
</config-property>
```

Select Load Balancing or Fail Over

The default behavior of CA Identity Manager is to use round-robin load balancing using the servers identified by the ConnectionURL and FailoverServers. Load balancing occurs if you leave FailOver set to false.

To select failover, set FailOver to true:

```
<config-property>
        <config-property-name>FailOver</config-property-name>
        <config-property-type>java.lang.String</config-property-type>
        <config-property-value>true</config-property-value>
</config-property>
```

Chapter 8: High Availability Provisioning Installation

Based on the guidelines in this chapter, you implement high availability for provisioning components by installing alternate Provisioning Servers and Provisioning Directories, and connector servers for C++ and Java connectors.

This section contains the following topics:

Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Basic installation ■ Installation on an application server cluster
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5. (Optional) Protect CA Identity Manager with SiteMinder.
X	**6. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.**

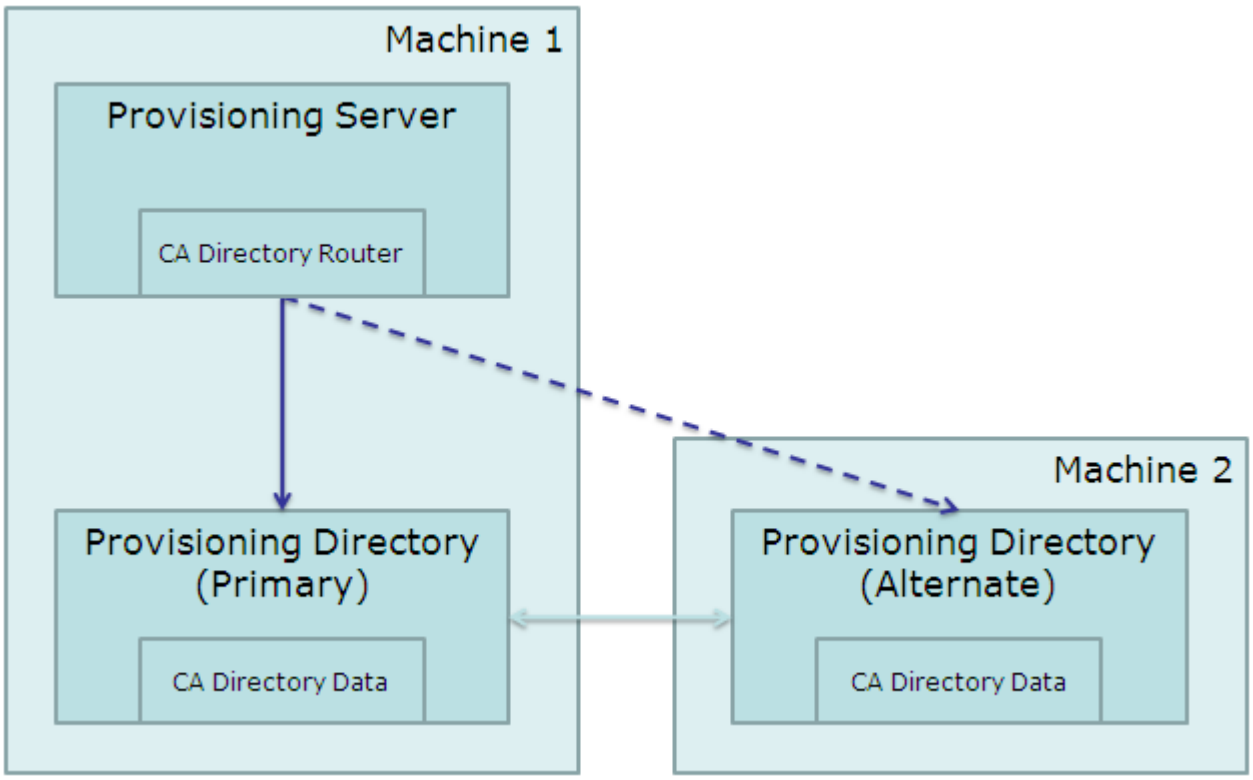
How to Install High Availability Provisioning Components

The following table describes the steps involved in installing provisioning components for high availability:

✓	Step
	1. Install primary and alternate Provisioning Servers and provisioning directories for load balancing and failover.
	2. Install several connector servers for load balancing and failover.
	3. Enable clients of the provisioning server to fail over.

Install Provisioning Directories

To support failover and load balancing, you can install primary and alternate Provisioning Directories. For example, you may have one system with the Provisioning Server on it and the primary Provisioning Directory. A second system has the alternate Provisioning Directory. If the primary Provisioning Directory fails, the alternate Provisioning Directory is assigned automatically.



You install alternate Provisioning Directories if they were not configured during the installation.

To install Provisioning Directories

1. Install the primary Provisioning Directory using the Provisioning Directory installer from where you unpacked the install package.

   - **Windows:**
     Unpacked-Install-Package\Provisioning\Provisioning Directory\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ProvisioningDirectory/setup

   If you have already installed a primary Provisioning Directory during the installation, you can omit step 1.

2. Perform the prerequisite configuration for the new Provisioning Directories.
3. Install one or more alternate Provisioning Directories.

Perform Prerequisite Configuration for New Provisioning Directories

You use the High Availability Configuration command before you use the Provisioning Directory installation program.

To Perform Prerequisite Configuration for New Provisioning Directories

1. Log into the system where the primary Provisioning Directory is installed.
2. On a command line prompt, navigate to the highavailability sub-directory where you unpacked the install package. For example:

   Unpacked-Install-Package\Provisioning\Provisioning Directory\highavailability
3. Enter this command:
   `highavailability.bat`

   The command displays a summary of the current configuration: the domain name, the hostname of each Provisioning Server and Provisioning Directory, and which one is the Primary Provisioning Directory.
4. Respond to the prompts to provide the hostnames required for each alternate Provisioning Directory that you want to add.

   If you plan to install alternate Provisioning Servers, you can add their hostnames now by responding to the prompts.
5. Log into all other Provisioning Directory and Provisioning Servers and repeat steps 2 through 4.

Install Alternate Provisioning Directories

Once you have performed the prerequisite configuration required, you can install alternate Provisioning Directories.

To install alternate Provisioning Directories

1. Log as a Local Administrator (for Windows) or root (for Solaris) into the system where you plan to install the alternate Provisioning Directory.
2. Make sure that CA Directory is installed on this system.
3. Copy custom schema files to the %DXHOME%/config/schema directory if any of the following is true for the primary Provisioning Directory:
   - COSX (etrust_cosx.dxc) has been modified
   - LDA connector (etrust_lda.dxc) is installed
   - A custom C++ connector schema has been created

   The Provisioning Directory installation checks the %DXHOME%/config/schema directory for extra schema files named etrust_*.dxc, and adds them to the group schema file, impd.dxg. If the custom schema files are not copied locally, data replication between the Provisioning Directories will fail.
4. Run the Provisioning Directory installer from where you unpacked the install package.
   - **Windows:**
     Unpacked-Install-Package\Provisioning\Provisioning Directory\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ProvisioningDirectory/setup

5. Select High Availability, and respond to the questions about the hostnames for systems where other Provisioning Directories are installed and which system is the primary Provisioning Directory.
6. Respond to other questions using the same answers given during the primary Provisioning Directory installation for:
   - Deployment Size
   - Shared Secret
   - FIPS key
7. Respond to this question based on how and when you want to replicate data from the Primary Provisioning Directory :

   ```
   Do you want to start replication to the Provisioning Directory.
   ```

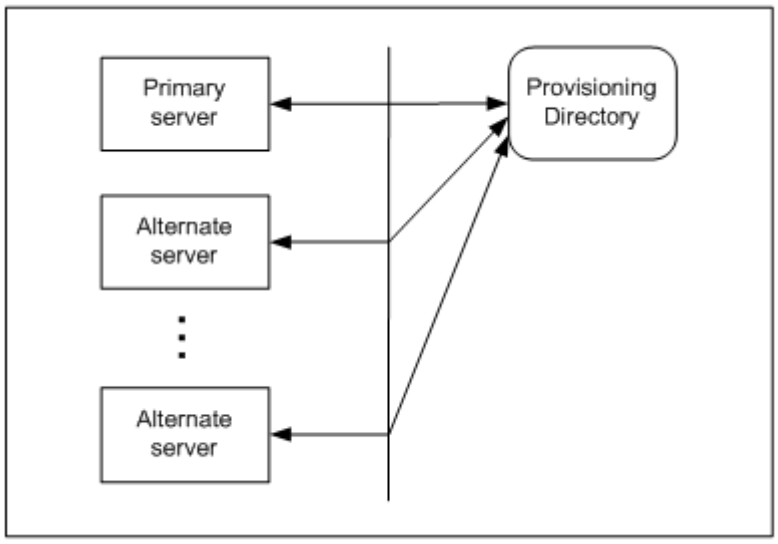
   If you are upgrading from a previous release, you may have a significant amount of data to replicate. You should deselect the checkbox if you do not want replication to start at this time. After the installation, you would then need to copy an LDIF data dump or online backup files from an existing Provisioning Directory and load the data or start the DSAs manually, which will start automatic replication.

Important! If alternate Provisioning Directory installation failed, data replication may have occurred before the failure. If so, the master and alternate Provisioning Directories have a record that replication occurred. If you now reinstall the alternate Provisioning Directory, that data is not replicated again. Instead, use the High Availability Configuration command on the primary and alternate Provisioning Directories to remove and add back the alternate Provisioning Directory before you reinstall it.

Provisioning Servers

Multiple Provisioning Servers share the workload of a provisioning domain, providing performance, scalability, and high availability. The first Provisioning Server installed is called the primary Provisioning Server. Additional servers are called alternate Provisioning Servers.

As shown in this illustration, you can configure multiple alternate Provisioning Servers for one primary Provisioning Server.



In this illustration, three Provisioning Servers are configured to serve the provisioning domain. All servers are configured to use the Provisioning Directory of the primary Provisioning Server installation.

Router DSA for the Provisioning Server

The Provisioning Server goes through a router DSA, and not directly to the Provisioning Directory. The router DSA, imps-router, is installed with the Provisioning Server installer. This DSA accepts requests from the Provisioning Server and routes them to the appropriate Provisioning Directory DSA (impd-co, impd-main, impd-inc, or impd-notify) depending on the prefix.

In a high-availability installation, the imps-router DSA has connection information for Provisioning Directory DSA on at least one alternate Provisioning Directory system. If a primary Provisioning Directory DSA becomes unavailable, the router DSA attempts to use an alternate DSA.

The imps-router DSA has been assigned ports 20391, 20391, 20393 (for address, SNMP, and console respectively).

Note:In previous releases of this software, the etrustadmin DSA used port 20391. Any connections to 20391 on the Provisioning Directory system fail unless the Provisioning Directory and Provisioning Server are on the same system. Therefore, reroute these connections to port 20391 on the Provisioning Server system.

For CA Directory DSAs running on one system to communicate with DSAs on another system, they must have connection information for each other. So during Provisioning Directory installation, you identify each Provisioning Server that can connect to it.

Install Provisioning Servers

To support failover, you can install primary and alternate Provisioning Servers. If you have already installed a Provisioning Server, you can omit step 1.

To install Provisioning Servers

1. Install the primary Provisioning Server using the Provisioning Server installer from where you unpacked the install package.
   - **Windows**:
     Unpacked-Install-Package\Provisioning\Provisioning Server\setup.exe
   - **UNIX**:
     Unpacked-Install-Package/Provisioning/ProvisioningServer/setup
2. Perform prerequisite configuration for the new Provisioning Servers.
3. Install one or more alternate Provisioning Servers.
4. Enter the alternate Provisioning Server host and port number when you enable provisioning in the Identity Manager Management Console. For details, see the *Configuration Guide*.

Perform Prerequisite Configuration for New Provisioning Servers

To configure knowledge files, you use the High Availability Configuration command on each system with a Provisioning Directory.

To Perform Prerequisite Configuration for New Provisioning Servers

1. Log into the system where the primary Provisioning Directory is installed.
2. On a command line prompt, navigate to the highavailability sub-directory where you unpacked the install package. It is a sub-directory of where you install the Provisioning Directory or Provisioning Server. For example:

   ```
   cd C:\\Program Files\Provisioning Directory\highavailability
   ```

3. Enter this command:

   ```
   highavailability.bat
   ```

   The command displays a summary of the current configuration: the domain name, and the hostname of each Provisioning Server and Provisioning Directory.
4. Respond to the prompts to provide the hostnames required for each Provisioning Server that you want to add.

   If you plan to also install alternate Provisioning Directories, you can add their hostnames now by responding to the command prompts.
5. Log into each system that will host a Provisioning Directory and repeat steps 2 through 4.

Install Alternate Provisioning Servers

Once you have performed the prerequisite configuration involving the highavailability command, you can install one or more Provisioning Servers.

To install alternate Provisioning Servers

1. Log in as a Local Administrator (for Windows) or root (for Solaris) on each system that will host an alternate Provisioning Server.
2. Make sure that CA Directory is installed on this system.
3. Copy custom schema files to the %DXHOME%/config/schema directory if any of the following is true for the primary Provisioning Directory:
   - COSX (etrust_cosx.dxc) has been modified
   - LDA connector (etrust_lda.dxc) is installed
   - A custom C++ connector schema has been created

   The Provisioning Directory installation checks the %DXHOME%/config/schema directory for extra schema files named etrust_*.dxc, and adds them to the group schema file, impd.dxg. If the custom schema files are not copied locally, the Provisioning Server will not route any custom schema.
4. Run the Provisioning Server installer from where you unpacked the install package.
   - **Windows**:
     Unpacked-Install-Package\Provisioning\Provisioning Server\setup.exe
   - **UNIX**:
     Unpacked-Install-Package/Provisioning/ProvisioningServer/setup
5. Complete the instructions in the installer dialog boxes.

   You can select a check box during installation to configure Provisioning Directory high availability. If you choose this option, you must supply the hostnames of any alternate Provisioning Directories and specify the primary Provisioning Directory.

Configure Provisioning Server Failover

For CA Identity Manager to distinguish the primary from the alternate Provisioning Server, you create server definitions in JIAM in the Management Console. You create these definitions in the directory object associated with the Identity Manager directory for your environment. During initialization, CA Identity Manager reads any failover server definitions defined in that object, adding them to the JIAM failover server definitions.

Note: For details on setting up server definitions, see the *Configuration Guide*.

Connector Servers

With the Connector Server Framework (CSF), you can run multiple Connector Servers and configure the Provisioning Servers to communicate with Connector Servers in specific contexts.

As a result, the Provisioning Server can:

- Support Connector Servers on different platforms to manage endpoint types that are unavailable on the platform where the Provisioning Server is installed.
- Communicate with multiple Connector Servers, which each manage a different set of endpoint types or endpoints. Therefore, endpoint types or endpoints can be managed on a parallel basis to achieve load balancing.

Connector Server Framework

The use of several Connector Servers is called the Connector Server Framework. The Connector Server Framework has two important characteristics:

- Scalability - multiple connector servers may share the load of working on a set of endpoints.

  For example, a lengthy exploration of an endpoint on one connector server does not influence the ability to operate on an endpoint that is being controlled by another Connector Server

- Communication channel security - communication between Provisioning Server and connector server is encrypted using TLS.

  If an endpoint type uses a proprietary protocol to communicate between the connector server and endpoints of that protocol, the extent of use of the proprietary protocol may be limited to a local network, or even to just local communication inside one server.

When deciding on an implementation strategy, consider these factors so that you protect the Connector Servers in your organization against unauthorized access:

- The Connector Server may be configured to disclose passwords in clear text.

  Any person with access to the system running the Connector Server and with sufficient privileges to modify the configuration of the Connector Server and to restart the Connector Server can make the Connector Server log passwords appear in clear text.

  The Connector Server is based on the open source slapd process. The instructions to make a slapd process log incoming passwords in clear text are in the public domain, for example, by looking at the manual pages at http://www.openldap.org

- The Connector Server is only protected by a bind password.

  The Connector Server trusts any client who connects to it and is able to provide the proper credentials, such as Bind DN and Bind Password. The Connector Server does not know if the connection comes from a Provisioning Server or not. Any user with internal access may disclose the bind password, then connect to the Connector Server from another server, and so have administrator privileges over the endpoints controlled by the Connector Server.

- The Connector Server is not protected against brute force attacks on the bind password

  Unlike the Provisioning Server, the Connector Server is not protected against repeated attempts at binding with different passwords. An attacker may therefore try to guess the password by brute force attack. Should an attacker succeed in guessing the bind password, then the road is open for the attacker to control the endpoints under control of the Connector Server.

For these reasons you are advised to design your implementation such that

- The same organizational unit is responsible for administrative access to all Provisioning Servers and connector servers.
- Your connector servers are suitably protected by firewalls or similar such that the ports may not be reached by unauthorized means.
- The ability to connect to Provisioning Servers and connector servers on non-TLS ports should be disabled in your production environments.

Load-Balancing and Failover

Failover and load-balancing of connector requests is achieved by each provisioning server based on the CSF configuration defined using csfconfig or Connector Xpress.

Each provisioning server consults the CSF configuration that applies to it and determines which Connector Servers it should use to access each endpoint or endpoint type. Failover and load-balancing occur when there are multiple connectors servers configured to serve the same endpoint or endpoint type.

Failover and load-balancing are unified and cannot be controlled separately. One cannot indicate that a particular connector server is to remain idle except when needed for failover. Instead, a provisioning server that is configured to use two or more connector servers interchangeably will distribute work between these connector servers (load balancing) during normal operation. Should one or more of the Connector Server become unavailable, the remaining connector servers will provide failover support for the unavailable connector servers.

Reliability and Scalability

With the Connector Server Framework (CSF), the Connector Server high availability features increase reliability and scalability.

Reliability is enhanced by having multiple Connector Servers serve a Provisioning Server, so it can continue to function if one or more Connector Servers become unavailable.

For example, if one Connector Server manages the UNIX endpoint type and another manages the Active Directory endpoint type; and the Active Directory Connector Server becomes unavailable, the Provisioning Server can still manage the UNIX endpoint types.

Scalability is achieved by having a mechanism to add more Connector Servers to manage an increasing amount of endpoint types or endpoints. For example, if the number of endpoint types increases to 100, the Provisioning Server can be configured to have 20 Connector Servers, with each Connector Server managing five endpoint types. Or configure 20 Connector Servers with each Connector Server managing overlapping sets of 10 endpoint types to allow for failover and load balancing behaviors as well.

Multi-Platform Installations

The Connector Server Framework is the configuration of Connector Servers that exist on multiple systems, which could be Windows or Solaris systems.

The following use cases are supported:

- Use Case 1
  - Provisioning Server and connector server installed on a Solaris system.
  - A second Connector Server installed on a Windows system, serving the non-multi-platform connectors.
- Use Case 2
  - Provisioning Server and connector server installed on a Windows system.
  - A second Connector Server installed on Solaris system, serving the multi-platform connectors.
  - A third Connector Server installed on a remote Windows system, serving the other connectors.

- Use Case 3
  - Provisioning Server installed on a Windows or Solaris system and a Connector Server installed on the same system.
  - Multiple additional Connector Servers installed on Windows or Solaris systems, serving as endpoint agents. This scenario is important for cases where the connector is using a proprietary or un-secured communication channel. Using this topology, the important segment of network traffic is secured by the standard Provisioning Server to Connector Server communication protocol and not by the proprietary protocol.

Install Connector Servers

Based on the guidelines in this chapter, you make connector servers highly available by installing several instances of Java Connector Servers or C++ Connector Servers, or both.

To install the Java Connector Server

If you plan to install more than one Java Connector Server, see the *Java Connector Server Implementation Guide* for additional guidelines. For a single Java Connector Server, follow these steps:

1. Run the following program where you unpacked the install package.
   - **Windows:**
     Unpacked-Install-Package\Provisioning\Connector Server\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ConnectorServer/setup
2. Complete the instructions in the installer dialog boxes.

To install the C++ Connector Server

1. Run the following program where you unpacked the install package.
   - **Windows:**
     Unpacked-Install-Package\Provisioning\Provisioning Server\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ProvisioningServer\setup.bin
2. Complete the instructions in the installer dialog boxes.

   This installation program also gives you the option to install alternate Provisioning Servers. However, for that component, a different procedure (see page 109) applies.

Configure Connector Servers

You configure the Connector Server Framework by using the csfconfig command or by using Connector Xpress. The csfconfig command uses the data in the Windows Registry (or UNIX counterpart created for Provisioning Server) to connect to a Provisioning Server. The csfconfig command must run on the system where one of the Provisioning Server runs.

Using the command, you can:

- Add or modify a Connector Server connection object with information such as the connector server, host, and port.
- Define for which endpoints or endpoint types the connector server is used; possibly varying this definition for alternate provisioning servers.
- Delete the Connector Server connection information object.
- List all connector server connection objects in a domain.
- Show one or all connector server connection objects for one or all connector servers

The csfconfig command uses the authorizations provided by a global user credential, so that global user must have the necessary administrative privileges to manipulate the appropriate ConfigParam and ConfigParamContainer objects.

csfconfig Command

To use the csfconfig command, the command line syntax is:

```
csfconfig [--help[=op]] [operation] [argument]
```

You can use these arguments in any order. The operation argument is required unless you are using the --help argument.

The --help[=op] option provides minimal on-line help. The “=op” argument may be used to list the arguments that are required or optional for the operation. For example, “--help=add” will provide a description of the add operation, while “--help” will provide general information.

If help is requested, other arguments are ignored and no request is sent to the server.

Note: The domain parameter can be omitted as it is always the domain used in the whole installation.

The following operations are available.

add

Add a new CS connection object. A name will be generated by this operation if one is not specified by the user. Required arguments: auth, host, pass. Optional arguments: authpwd, br-add, desc, domain, name, port, usetls, debug.

addspec

Adds a branches specialization for one provisioning server.

When you have installed alternative provisioning servers, sometimes a connector server is not to be used by all of these Provisioning Servers. Or sometimes different provisioning servers will want to use the same connector servers for different branches (endpoint types or endpoints). A branches specialization is a list of branches that is specific to one provisioning server. Only provisioning servers without a specialization will use the branches specified in the main CS connection object. Required arguments: auth, name, server. Optional arguments: authpwd, br-add, domain, debug.

list

List all CS connection objects. Required arguments: auth. Optional arguments: authpwd, domain, debug.

modify

Modify a CS connection object. Required arguments: auth, name. Optional arguments: authpwd, br-add, br-rem, desc, domain, host, pass, port, usetls, debug.

modspec

Edits a specialization created by addspec. Required arguments: auth, name, server. Optional arguments: authpwd, br-add, br-rem, domain, debug.

remove

Remove an existing CS connection object. Required arguments: auth, name. Optional argument: authpwd, debug.

remspec

Removes a specialization created by addspec. Required arguments: auth, name, server. Optional arguments: authpwd, domain, debug.

modify

Modify a CS connection object. Required arguments: auth, name. Optional arguments: authpwd, br-add, br-rem, desc, domain, host, pass, port, server, tls, usetls.

show

Show a specific CS connection object or show all CS connection objects. The output shows the host and port of the connector server if it is available. Required arguments: auth. Optional arguments: authpwd, name, domain, debug.

Each operation takes several arguments in the form “name=value”. Spaces are not allowed before or after the “=” symbol, and if the value contains any spaces, the argument must be quoted appropriately for the platform (Windows or UNIX). Except as noted, the value must be provided, and must be non-empty.

The following arguments are used for the operations as noted above:

auth=<value>

Identify the global user for authentication.

Value format: “name” where name is the global user's name.

authpwd=<value>

Identify a file containing just the global user's password on the first line. If this argument is not specified, the user will be prompted for a password.

Value format: any appropriate operating system file path.

br-add=<value>

Add a new branch. This argument may be specified multiple times to add multiple branches.

Value format: “[[endpoint,]endpoint type][@[domain]]”. Use a branch of “@” by itself to represent all branches. Add “endpoint type” or “endpoint,endpoint type” to identify a specific endpoint type or endpoint.

br-rem=<value>

Remove an existing branch. This argument may be specified multiple times to remove multiple branches.

Value format: same format as specified for br-add.

debug=<value>

Turns on trace logging for the command. Tracing messages are written to the file PSHOME\logs\etaclientYYYYMMDD.log file.

Value format: The value “yes" enables logging.

desc=<value>

Provide an arbitrary description for the object. If not specified in an add operation, it will default to the value of the host argument.

Value format: an arbitrary string.

domain=<value>

Define the default domain. If not specified, the domain specified in the auth argument is used as the default.

As the value can only be the default, this parameter can always be omitted

host=<value>

Define the name of the host on which the Connector Server runs.

Value format: any legal host name or IP address.

name=<value>

The name of the Connector Server object. If not specified during Add, csfconfig will assign a name and display what name was created.

Value format: A case-insensitive string of one or more characters consisting of upper-case English letters (A-Z), lower-case English letters (a-z), digits (0-9), hyphen(-) or underscore(_).

pass[=<value>]

Define the file containing the password for the Connector Server connection object. If the value is not specified, the user will be prompted.

Value format: any appropriate OS file path.

Important! The password you must specify is the password you entered when you installed that Connector Server or you changed subsequent to install by running the pwdmgr utility on that Connector Server system.

port=<value>

Define the port number for the object. This must be a valid number for a port where the Connector Server listens for connections.

Value format: an integer.

server[=<value>]

In addspec, modspec and remspec commands, define the name of the Provisioning Server that is served by the Connector Server . The branches defined in a specialization override, for a particular provisioning server, the branches defined in the CS configuration object by add and modify commands.

Value format: the name of the host where the Provisioning Server is running as returned by the system's hostname command.

Note: The Connector Server configuration objects are stored with the other domain configuration parameters in the provisioning directory. While the Connector Server configuration parameters cannot be viewed or changed with the provisioning manager directly, one can use the provisioning manager (System task, Domain Configuration button) to get a list of known provisioning servers. To do this, open the “Servers” parameter folder and the known provisioning servers will be listed.

usetls[=<value>]

Indicate if TLS should be used to communicate with the Connector Server. The value is optional for the add operation only, in which case it defaults to “yes.” .

Value format: a string “yes” or “no”.

Upon successful completion of the add operation, the name of the newly created Connector Server connection object will be listed. If the name parameter is missing, a name is generated. For example:

```
Created CS object with name = SA000
```

For most operations, successful or not, the status and a message (if any) will be shown. For example:

```
The host name, port number, or TLS flag was successfully changed. The branch settings
were successfully changed.
```

For some errors, such as invalid command line parameters, no status code or server error message is displayed. In these cases, a simple statement of the error will be shown. For example:

```
$ csfconfig add
No authentication information supplied.
For on-line help, use "--help [=<op>]"
```

csfconfig Command Examples

To specify that the UNIX and CA Access Control endpoint types should be served by the Connector Server running on host "sunserver01" and the remaining endpoint types served by a Connector Server running on host "windows02", issue the following commands.

Each command execution prompts you for the etaadmin password.

```
csfconfig add \
auth="etaadmin" \
br-add="UNIX – etc" \
br-add="UNIX – NIS-NIS plus Domains" \
br-add="Access Control" \
host="sunserver01" \
usetls="yes"

csfconfig add \
auth="etaadmin" \
br-add="@" \
host="windows02" \
usetls="yes"
```

C++ Connector Server on Solaris

The C++ Connector Server installed on Solaris can manage only Solaris UNIX ETC and ACC endpoints. For all other Connectors, install the C++ Connector Server on a Windows system and register it with the Provisioning Server installed on Solaris. During installation, specify that this Connector Server is your default C++ Connector Server.

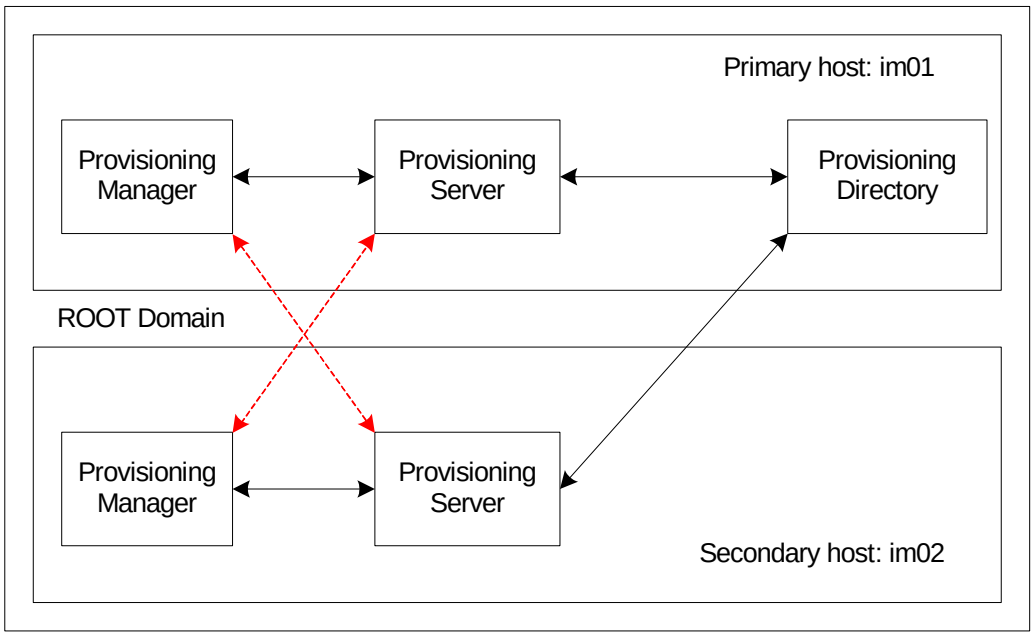
Failover for Provisioning Clients

Client-tier configuration includes the following tasks:

- Configure the Windows client-tier failover
- Configure the Provisioning Manager to communicate with their local Provisioning Servers, and fail over to the remote Provisioning Server

You use the same Provisioning Manager dialog to accomplish both of these tasks, on each server in turn.

The configuration shown in the following illustration lets Provisioning Manager submit identity provisioning requests to one Provisioning Server and fail over to another server:



The Provisioning Manager sends requests to the default Provisioning Server and fails over to another server.

Enable User Console Failover

If the application server for the Identity Manager Server fails, it does not receive Provisioning Server updates. As a result, the Identity Manager User Console does not show provisioning changes. Therefore, you should configure an alternate URL for the Identity Manager Server.

To enable the client-tier failover for the User Console

1. Launch the Provisioning Manager.
2. Click System, Identity Manager Setup.
3. Fill in the host name and port for another system in the cluster.
4. Fill in the environment.

   It must be the same one that is on the primary URL.
5. Click Add.

Enable Provisioning Manager Failover

You can enable Provisioning Manager failover on both the primary and secondary host servers. When this procedure is complete, you will have configured each server for failover to the other.

To enable the Provisioning Manager failover

1. Launch the Provisioning Manager.
2. Select File, Preferences, and select the Failover tab.
3. Mark the Enable Failover check box. By default, the local Provisioning Server is already defined.
4. Click Add.
5. Enter the host name of the remote Provisioning Server.

   For example, on im01, enter the server host for im02. On im02, enter the server host for im01.
6. Enter 20389 for the LDAP port value and 20390 for the LDAP/TLS port value, respectively.
7. Adjust the preference order by moving the entries up and down in the list.
8. Click OK.
9. Restart the Provisioning Manager to enable your changes.

Test the Provisioning Manager Failover

You can test your client failover configuration by performing the following procedure:

To test Provisioning Manager failover

1. Stop the CA Identity Manager - Provisioning Server service on one domain server.
2. Issue one or more operations using Provisioning Manager for this server installation.

   Since you stopped the CA Identity Manager - Provisioning Server service locally, the traffic flows to the failover domain server. If it does not, check your configuration and try the test again.

Appendix A: UNIX, Linux, and Non-Provisioning Installations

For UNIX and LINUX systems and scenarios where no provisioning software is needed, some additional instructions apply.

This section contains the following topics:

UNIX and Console Mode Installation

The examples in this guide provide the Solaris executable name for the installation program. However, you may be installing on AIX or Linux.

- For AIX, use: ca-im-*release*-aix.bin
- For LINUX, use: ca-*release*-linux.bin

release represents the current release of CA Identity Manager

If you are performing an installation in console mode, such as on a UNIX workstation, you add another option to the command line.

- For the main installation, add -i console. For example:
  `./ca-im-12.5-sp*N*-sol.bin -i console`
- For installation of provisioning components, add -console to the setup command.

Red Hat Linux 64-bit Installation

If you plan to install CA Identity Manager on a Red Hat Linux 64-bit system, you need to create a symbolic link to work around a CryptoJ failure. Create a link as follows:

```
ln –s /dev/urandom /dev/random
```

Non-Provisioning Installation

This guide refers to the Windows and Solaris program names for the installers that provide options to install provisioning software. If you prefer to see no provisioning options, you can use these installers:

- For Windows, use IMWithoutProvisioning\ca-im-web-*release*-win32.bat
- For Solaris, use IMWithoutProvisioning/ca-im-web-*release*-sol.sh

release represents the current release of CA Identity Manager.

Appendix B: Uninstallation and Reinstallation

This section contains the following topics:

How to Uninstall CA Identity Manager

To fully uninstall CA Identity Manager, remove CA Identity Manager software components and clean up the CA Identity Manager-specific configuration in your application server. The following checklist describes the steps to uninstall CA Identity Manager:

✓	Step
	1. Delete CA Identity Manager objects with the Management Console.
	2. (Optional) If you used SiteMinder, remove the CA Identity Manager schema from the policy store or remove the Policy Server. For more information, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide*.
	3. Use the highavailability command to uninstall Provisioning Directories and Provisioning Servers from this location: *Unpacked-Install-Package*\Provisioning\Provisioning Directory\highavailability
	4. Uninstall the CA Identity Manager components.
	5. Remove CA Identity Manager configuration information from the application server.

Remove CA Identity Manager Objects with the Management Console

In order to remove objects created automatically by CA Identity Manager when you configure environments and directories, use the Management Console.

1. Open the Management Console:

   http://*im_server*:*port*/iam/immanage

2. Click Environments.
3. Select all of the check boxes for the existing Environments.
4. Click Delete.
5. Click Directories.
6. Select all of the check boxes for the existing Directories.
7. Click Delete.

Remove the CA Identity Manager Schema from the Policy Store

If you were using a SiteMinder Policy Server, remove the CA Identity Manager schema from the policy store.

Remove the CA Identity Manager schema from a SQL Policy Store

On systems where you installed the CA Identity Manager Extensions for SiteMinder, remove the CA Identity Manager schema. The default location for the command to remove the schema follows:

- SQL Server:

  ```
  C:\Program Files\CA\Identity Manager\IAM Suite\Identity
  Manager\tools\policystore-schemas\mssql\
  ims8_mssql_ps_delete.sql
  ```

- Oracle:

  ```
  /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/policystore-schemas/
  oracle/ims8_oracle_ps_delete.sql
  ```

Remove the CA Identity Manager schema from an LDAP Policy Store

Note: If you are using Microsoft Active Directory or Microsoft ADAM as a policy store, you do not need to complete this procedure. You cannot remove schema objects from these policy stores. However, you can disable them. For more information, see the documentation for your directory.

To remove the CA Identity Manager schema from an LDAP policy store

1. Complete one of the following:
   - If you are using IBM Directory Server as a policy store, in the IBM Directory Server Web Administration user interface, remove the schema file V3.imsschema60 from the Files section of the schema configuration. Then, restart the directory server.

     Note: There are no other steps required to remove the schema from an IBM Directory Server. Continue with Uninstall CA Identity Manager Software Components.
   - If you are using CA Directory as a policy store, remove the etrust_ims.dxc file from *dxserver_home*\config\schema.

     where *dxserver_home* is the install location of CA Directory.

     Note: There are no other steps required to remove the schema from a CA Directory Server. Continue with Uninstall CA Identity Manager Software Components.
   - If you are using another LDAP directory as a policy store, skip to Step 2.
2. Navigate to the policystore-schemas folder. These are the default locations:
   - **Windows:** C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas
   - **UNIX:** /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/policystore-schemas
3. Use the appropriate LDIF schema file from the following table to remove the schema from the directory.

   Note: For more information on removing schema files, see the documentation for your directory.

Directory Type	LDIF File
Novell eDirectory	novell\novell-delete-ims8.ldif
Oracle Internet Directory (OID)	oracle-internet-directory\oracle-internet-directory-delete-ims8.ldif

Directory Type	LDIF File
Sun Java Systems (Sun One, iPlanet)	sunone\sunone-delete-ims8.ldif

Uninstall CA Identity Manager Software Components

Use the instructions in this section to uninstall CA Identity Manager components from each system on which you installed a component. For example, if you installed the Identity Manager Server and the Identity Manager Administrative Tools on separate systems, uninstall components from both systems.

To uninstall CA Identity Manager software components on Windows

1. Install a 32-bit JRE or JDK, which is required for the uninstallation program to run.
2. Go to Start, Control Panel, Add/Remove Programs and select CA Identity Manager.
3. Select CA Identity Manager.
4. Click Change/Remove.

   All non-provisioning components will be uninstalled.
5. For any provisioning components, use the individual component installer to uninstall the component.

   Note: Although you install Provisioning Manager with Administrative Tools, you use the Provisioning Manager installer to uninstall this component.

To uninstall CA Identity Manager software components on UNIX

1. Navigate to the following location:

   /opt/CA/Identity_Manager/install_config_info/im-uninstall/uninstall
2. Run the following script:

   ```
   sh im-uninstall.sh
   ```

   Follow the on-screen instructions.
3. For any provisioning components, use the individual component installer to uninstall the component.

Remove CA Identity Manager from WebSphere

After uninstalling CA Identity Manager software, you can remove the CA Identity Manager configuration from your application server by using the WebSphere Administrative Console or by executing scripts from the command line.

To remove CA Identity Manager using the Administrative Console

1. Open the WebSphere Administrative Console using the following URL:

   http://*websphere_server*:9060/admin

2. Select Applications, Enterprise Applications.
3. In the Enterprise Applications screen, select the check box next to CA Identity Manager and click Stop.
4. Select the check box next to CA Identity Manager and click Uninstall.
5. If you installed the SiteMinder EAR and SiteMinder Agent EAR, stop these applications, and uninstall them as described previously.
6. Click Save.
7. Click Save to save changes to the master configuration.
8. Remove the ca-stylesr5.1.1.ear file.

   Note: Only remove the ca-stylesr5.1.1.ear if no other CA product is using it.

To remove CA Identity Manager using the command line

CA Identity Manager includes two scripts that you can use to remove CA Identity Manager from the WebSphere application server:

- Uninstall script (uninstallApp.jacl)—Stops the CA Identity Manager application, then removes it from WebSphere.
- Cleanup script (Ims6Cleanup.jacl) —Removes the CA Identity Manager resources, such as those created by running the uninstallApp.jacl.

  Note: Running the Cleanup script removes resources that are used by all CA Identity Manager installations running on the same application server. If you have CA Identity Manager installations on the same system that you do not want to delete, you should not run the Cleanup script. Also, this script does *not* remove any data sources created by CA Identity Manager.

To remove CA Identity Manager using the command line, perform the following procedure.

1. From the command line, navigate to *websphere_home*\bin.
2. Be sure that the WebSphere application server is running.
3. Run the Uninstall script as follows:
   - **Windows:** wsadmin -f uninstallApp.jacl
   - **Unix:** ./wsadmin.sh -f uninstallApp.jacl

4. Run the Cleanup script as follows:
   - **Windows:** wsadmin -f Ims6Cleanup.jacl *websphere_node*
   - **Unix:** ./wsadmin.sh -f Ims6Cleanup.jacl *websphere_node*

   where *websphere_node* is the name of the WebSphere node where CA Identity Manager was installed.
5. Remove the ca-stylesr5.1.1.ear file.

   Note: Only remove the ca-stylesr5.1.1.ear if no other CA product is using it.
6. Remove the service integration bus as follows:
   a. In the WebSphere Administrative Console, go to Service Integration, Buses.
   b. Remove IMSBus.
   c. Stop the application server.
   d. Remove the *node_name.server_name*.IMSBus directory under *websphere_home*\profiles*websphere_profile*\databases\com.ibm.ws.sib\

Reinstall CA Identity Manager

You can reinstall any of the CA Identity Manager software components by rerunning the installer. When you run the installer, it detects any CA Identity Manager components installed on the system. You may reinstall the same components that you originally installed on the system or other components that were not originally on the system.

Note: Reinstalling the Identity Manager Administrative Tools replaces all of the files in the Administrative Tools directory. To prevent overwriting custom files, back up the directory where the Administrative Tools are installed.

Appendix C: Unattended Installation

This section contains the following topics:

How to Run an Unattended Installation

To run the installer in the unattended installation mode

1. Modify the im-installer.properties file.
2. Run the following command:
   - **Windows:**
     `ca-im-12.5-sp01-win32.exe -f im-installer.properties -i silent`
   - **UNIX:**
     `./ca-im-12.5-sp01-sol.bin -f im-installer.properties -i silent`

Modify the Configuration File

To enable an unattended CA Identity Manager installation, modify the settings in the im-installer.properties configuration file using a text editor. The default parameters in the file reflect the information entered during the initial CA Identity Manager installation. Change the default values as needed.

Note the following when modifying the configuration file:

- Make a back-up copy of the installer properties file before modifying the original, since the file holds all of the values you entered during the initial installation or configuration.
- Do not add extra spaces between the parameter name, the equals sign (=), and the attribute value.
- All directory names on Windows must contain either double back slashes or forward slashes, not single back slashes.

Initial Choices

For basic installation choices, enter values for the following parameters:

Parameter	Instructions
DEFAULT_NEW_INSTANCE_DISPLAY_NAME	Enter 'New Installation' if this is a fresh install. For upgrades, this will be blank.
DEFAULT_COMPONENTS	Enter one or more components: ■ Server - Identity Manager Server ■ Exten - Extensions to the Policy Server ■ Admin - Identity Manager Administrative Tools ■ Provision - Provisioning Server ■ Directory - Provisioning Directory To install more than one component, separate components by a comma.
DEFAULT_INSTALL_FOLDER	Enter the directory in which to install the Identity Manager Server.
DEFAULT_GENERIC_USERNAME	Generic login information for CA Identity Manager components that are installed.
DEFAULT_GENERIC_PASSWORD	Generic password information for CA Identity Manager components that are installed.
DEFAULT_FIPS_MODE	Select if you want to enable FIPS 140-2 compliance.
DEFAULT_FIPS_KEY_LOC	Enter the path to the FIPS key location.

The installation program ignores any parameters that do not apply to the component you are installing. For example, if you set DEFAULT_COMPONENTS to Exten, only the DEFAULT_PS_ROOT and DEFAULT_USE_SITEMINDER parameters are used.

Identity Manager Server

If you plan to install the Identity Manager Server, enter values for the following:

Parameter	Instructions
DEFAULT_APP_SERVER	Enter, Weblogic, WebSphere, or JBoss
DEFAULT_APP_SERVER_URL	Enter full URL of the application server hosting CA Identity Manager, including the port.
DEFAULT_JAVA_HOME	Path to JRE or JDK for CA Identity Manager.
Additional Database Parameters	
DEFAULT_DB_HOST	Enter the hostname of the system hosting the CA Identity Manager database.
DEFAULT_DB_PORT	Enter the port of the system hosting the CA Identity Manager database.
DEFAULT_DB_NAME	Enter the name of the CA Identity Manager database.
DEFAULT_DB_USER	Enter the administrative username for the CA Identity Manager database.
DEFAULT_DB_PASSWORD	Enter the password for the administrative user of the CA Identity Manager database.
DEFAULT_DB_TYPE	Enter the type of database used for the CA Identity Manager database.
Additional JBoss Parameter	
DEFAULT_JBOSS_FOLDER	Enter the full pathname of the directory where you installed the JBoss application server. For example, C:\jboss-4.2.3
Additional WebLogic Parameters	
DEFAULT_BINARY_FOLDER	Enter the full directory path of the directory where you installed WebLogic. For example: C:\bea\weblogic92\

Parameter	Instructions
DEFAULT_DOMAIN_FOLDER	Enter the full path and directory name for the WebLogic domain you created for CA Identity Manager.
DEFAULT_SERVER_NAME	Enter the name of the WebLogic server instance you created for use with CA Identity Manager.
DEFAULT_BEA_CLUSTER	Enter the cluster name for the WebLogic cluster.
Additional WebSphere Parameters	
DEFAULT_WEBSPHERE_FOLDER	Enter the full pathname of the directory where you installed CA Identity Manager Tools for WebSphere.
DEFAULT_WAS_NODE	Enter the name of the node in which the application server is located.
DEFAULT_WAS_SERVER	Enter the name of the system on which the application server is running.
DEFAULT_WAS_CELL	Enter the name of the cell in which the application server is located.
WAS_PROFILE	(WebSphere 6) Enter the location of the WebSphere profile files.
DEFAULT_WAS_CLUSTER	(WebSphere 6) Enter the cluster name for the WebSphere cluster.

If you are using a SiteMinder Policy Server, enter the following:

Parameter	Instruction
DEFAULT_PS_HOST	Enter the fully-qualified domain name of the Policy Server.

Parameter	Instruction
DEFAULT_PS_USER	Enter the user name of the Policy Server administrator.
DEFAULT_PS_PW	Enter the password of the Policy Server administrator.

Provisioning Components

If you install Provisioning, enter the following:

Parameter	Instruction
DEFAULT_CONFIG_REMOTE PROVISIONING	Enter true if you are connecting to a remote Provisioning Directory.
DEFAULT_DEPLOYMENT_SIZE	Enter the size of your Provisioning Directory deployment.
DEFAULT_DIRECTORY_IMPS_HOSTN AMES	Enter the hostnames of all Provisioning Servers that will connect to the Directory.
DEFAULT_DOMAIN_NAME	Enter 'im' unless you have an existing Provisioning domain.
DEFAULT_DIRECTORY_HOST	Enter the hostname of the system with Provisioning Directory installed.
DEFAULT_DIRECTORY_PORT	Enter the port number of the system with the Provisioning Directory installed.
DEFAULT_DIRECTORY_PASSWORD	Enter the password for the Provisioning Directory.

Extensions for SiteMinder

To install the extensions for a SiteMinder Policy Server, enter the following:

Parameter	Instruction
DEFAULT_PS_ROOT	(Solaris Only) Enter the directory where the Policy Server is installed.
DEFAULT_USE_SITEMINDER	Enter true if you are using a SiteMinder Policy Server in your implementation.

Configuration File Format

The im-installer.properties file is located in the CA Identity Manager installation directory. For example:

- **Windows**: C:\Program Files\CA\CA Identity Manager\install_config_info
- **UNIX**: /opt/CA/IdentityManager/install_config_info/im-installer.properties

The following is an example of the im-installer.properties file created during a CA Identity Manager installation:

```
############################################################
### Silent input properties file for the IMR12.5 installer ##
############################################################

#INSTANCE DISPLAY NAME
#   For fresh installation it will always be 'New Installation'
#   For Upgrade NEW_INSTANCE_DISPLAY_NAME will be equal to INSTANCE_NAME
#DEFAULT_NEW_INSTANCE_DISPLAY_NAME=

# Component list
#   Valid values (comma-separated, one or more):
Server,Exten,Admin,Provision,Directory
DEFAULT_COMPONENTS=

# Install folder
#   All products are installed in subfolders under this folder
#   This is parent product root selected by the user
#   For e.g. C:\\Program Files\\CA
DEFAULT_INSTALL_FOLDER=

#Generic login information
DEFAULT_GENERIC_USERNAME=
#DEFAULT_GENERIC_PASSWORD=<For silent install, insert generic user password here and
uncomment line.>

# Provisioning Server and Provisioning Directory Information.
#  Configure the Provisioning Server to a remotely installed Provisioning
Directory(true/false)
DEFAULT_CONFIG_REMOTE_PROVISIONING=

 #Select the deployment type that suits your needs (1,2,3 or 4): 1. Compact 2. Basic
3. Intermediate (64 Bit only) 4. Large (64 Bit only)
DEFAULT_DEPLOYMENT_SIZE=
DEFAULT_DIRECTORY_IMPS_HOSTNAMES=
DEFAULT_DOMAIN_NAME=
DEFAULT_DIRECTORY_HOST=
DEFAULT_DIRECTORY_PORT=
#DEFAULT_DIRECTORY_PASSWORD=<For silent install, insert password to be used with
Provisioning Components here and uncomment line.>
```

```
#FIPS 140-2 Compliance mode (true/false) for Identity Manager, Admin Tools,
Provisioning Manager and Provisioning Server
DEFAULT_FIPS_MODE=
#Complete path of the FIPS key file. For e.g. C:\\Program Files\\FIPSkey.dat
DEFAULT_FIPS_KEY_LOC=

#Identity Manager Application Server information
#   App Server
#   Valid values:  JBoss, WebLogic10, WebSphere6, WebSphere7
DEFAULT_APP_SERVER=
DEFAULT_APP_SERVER_URL=

#Path to JDK for the JBOSS Application Server. No input required for other Application
Servers
DEFAULT_JAVA_HOME=

#JBoss info
DEFAULT_JBOSS_FOLDER=

#Weblogic info
DEFAULT_BINARY_FOLDER=
DEFAULT_DOMAIN_FOLDER=
DEFAULT_SERVER_NAME=

#For Weblogic 9 & 10 only:
DEFAULT_BEA_CLUSTER=

#WebSphere info
DEFAULT_WEBSPHERE_FOLDER=

#WAS_NODE Value: $WAS_HOME$\installedApps\$WAS_NODE$ or
$WAS_HOME$\config\cells\$WAS_CELL$\nodes\$WAS_NODE$. These should be same.
DEFAULT_WAS_NODE=

#WAS_SERVER Value:
$WAS_HOME$\config\cells\$WAS_CELL$\nodes\$WAS_NODE$\servers\$WAS_SERVER$
DEFAULT_WAS_SERVER=

#WAS_CELL Value: $WAS_HOME$\config\cells\$WAS_CELL$
DEFAULT_WAS_CELL=

#WAS_PROFILE Value: $WEBPHERE_HOME$\profiles\$WAS_PROFILE$
WAS_PROFILE=

#WAS_CLUSTER Value: $WAS_HOME$\config\cells\$WAS_CELL$\clusters\$WAS_CLUSTER$
DEFAULT_WAS_CLUSTER=
```

```
#Policy Server info
DEFAULT_PS_HOST=
DEFAULT_PS_USER=
#DEFAULT_PS_PW=<For silent install, insert PS Admin user password here and uncomment
line.>

#8.1 Migration
# SiteMinder Agent Name
DEFAULT_AGENT_NAME=
# SiteMinder Shared Secret
DEFAULT_AGENT_PW=
# Automatically migrate. Valid values (true/false)
DEFAULT_MIGRATE_DIR_ENV=
# Directory to export to
DEFAULT_DIR_ENV_EXPORT=

#Policy Server Extensions info
#   Location of CsSmPs-<Instance name> folder
DEFAULT_PS_ROOT=
#You can use the SiteMinder Policy Server and a SiteMinder Web Agent to provide advanced
security
# for CA Identity Manager environments. Valid values (true/false)
DEFAULT_USE_SITEMINDER=

#Database Info
DEFAULT_DB_HOST=
DEFAULT_DB_PORT=
DEFAULT_DB_NAME=
DEFAULT_DB_USER=
#DEFAULT_DB_PASSWORD=<For silent install, insert database password here and uncomment
line.>

#Following are permissible values: mssql2005, or oracle10
DEFAULT_DB_TYPE=

#Upgrading from IM8.1sp2
#       If you have data stores located on separate servers or you wish to install
them on separate
#       servers, you can specify them below. Otherwise if you wish to have all the
data stores on
#       the same server, change the DEFAULT_DB_* properties from above.

#Object Store Datastore Info
#DEFAULT_OS_DB_HOST=
#DEFAULT_OS_DB_PORT=
#DEFAULT_OS_DB_NAME=
#DEFAULT_OS_DB_USER=
#DEFAULT_OS_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>
```

```
#Task Persistence Datastore Info
#DEFAULT_TP_DB_HOST=
#DEFAULT_TP_DB_PORT=
#DEFAULT_TP_DB_NAME=
#DEFAULT_TP_DB_USER=
#DEFAULT_TP_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Audit Datastore Info
#DEFAULT_AUDIT_DB_HOST=
#DEFAULT_AUDIT_DB_PORT=
#DEFAULT_AUDIT_DB_NAME=
#DEFAULT_AUDIT_DB_USER=
#DEFAULT_AUDIT_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Reporting Snapshot Datastore Info
#DEFAULT_RS_DB_HOST=
#DEFAULT_RS_DB_PORT=
#DEFAULT_RS_DB_NAME=
#DEFAULT_RS_DB_USER=
#DEFAULT_RS_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Workflow Datastore Info
#DEFAULT_WF_DB_HOST=
#DEFAULT_WF_DB_PORT=
#DEFAULT_WF_DB_NAME=
#DEFAULT_WF_DB_USER=
#DEFAULT_WF_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

# Automatically Upgrade Workflow DB
DEFAULT_UPGRADE_WF_DB=

# Automatically Migrate Task Persistance
DEFAULT_MIGRATE_TP=
```

Appendix D: Installation Log Files

The log files are stored based on where you unpack the installation package. The following examples may have different top-level directories than these default locations.

This section contains the following topics:

Log Files on Windows

If you encounter any issues while performing a CA Identity Manager installation, see the caiamsuite.log file in this location:

C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager

The Identity Manager Server installer logs are written to the following default location:

C:\Program Files\CA\Identity Manager\install_config_info

On a 64-bit windows system, the default location is:

C:\Program Files (x86)\CA\Identity Manager\install_config_info

The Provisioning installer logs are written to the user's Temp directory.

Example:

C:\Documents and Settings*user*\Local Settings\Temp\imps_server_install.log

Log files on UNIX

If you encounter any issues while performing a CA Identity Manager installation, see the caiamsuite.log file in this location:

/opt/CA/IdentityManager/

The Identity Manager Server installer logs are written to the following default location:

/opt/CA/IdentityManager/install_config_info

The Provisioning installer logs are written to the user's Temp directory.

Appendix E: Windows Services Started by CA Identity Manager

The following are the services started on Windows when you install and start all components of CA Identity Manager:

- CA Directory impd-co
- CA Directory impd-inc
- CA Directory impd-notify
- CA Directory impd-router
- CA Directory SSL Daemon – impd
- CA Identity Manager Connector Server (C++)
- CA Identity Manager Connector Server (Java)
- CA Identity Manager Provisioning Server
- Enterprise Common Services (Transport)
- Enterprise Common Services GUI Framework
- Enterprise Common Services Store-And-Forward Manager

This list of services may useful to you for troubleshooting purposes.

Appendix F: Installation Checklists

Use the following checklists in this appendix in the order they appear to help you install and configure CA Identity Manager. You may want to print the checklists and check off the steps as you complete them.

This section contains the following topics:

How to Install Prerequisite Components

To install the prerequisite hardware and software for CA Identity Manager:

✓	Step
	1. Make your system meet the hardware and software requirements.
	2. Create a database.
	3. Set up the application server as required.
	4. Fill in the Installation Worksheets with information you need to supply during the CA Identity Manager installation.

How to Perform a Basic Installation

Use the following checklist to perform a basic installation of CA Identity Manager:

✓	Step
	1. Install CA Identity Manager components on the systems required.
	2. Add support for SiteMinder if you are installing on a 64-bit AIX system.

✓	Step
	3. Verify the Identity Manager Server starts.
	4. Configure Provisioning Manager if installed on a remote system.
	5. Install optional provisioning components.

How to Install CA Identity Manager on a WebSphere Cluster

The following procedures describe how to install CA Identity Manager on a WebSphere cluster.

✓	Step
	1. Run the Installation from the Deployment Manager (see page 49)
	2. Add SiteMinder Support on 64-bit AIX (see page 51)
	3. Export and Import the EAR on 64-Bit AIX Systems (see page 53)
	4. Add Cluster Members (see page 54)
	5. Configure Messaging Engines (see page 54)
	6. Create Message Stores (see page 55)
	7. Create Core Group Policies (see page 56)
	8. Configure Workflow for WebSphere (see page 57)
	9. Update the WebSphere Path for SiteMinder (see page 58)
	10. Configure the Proxy Plug-In for the Web Server (see page 58)

How to Create Separate Databases

To create separate databases for CA Identity Manager:

✓	Step
	1. Create a MS SQL Server or Oracle database instance for CA Identity Manager.
	2. Edit the data source.

✓	Step
	3. (Optional) Run the SQL scripts.

How to Install the Report Server

The following checklist describes the steps to install CA Identity Manager's reporting feature:

✓	Step
	1. Review the report pre-installation checklist.
	2. Gather reporting information.
	3. Open ports required by the Report Server.
	4. Install the Report Server (CA Business Intelligence)
	5. Run the Registry Script.
	6. Copy the JDBC JAR files.
	7. Deploy the default reports.

Note: For more information on configuring reporting after the installation, see the *Administration Guide*.

How to Protect CA Identity Manager with SiteMinder

The following table describes the steps involved in configuring SiteMinder to protect CA Identity Manager resources:

✓	Step
	1. Be sure you have installed the Identity Manager extensions on the SiteMinder Policy Server as described in the Installation Prerequisites chapter.
	2. Install a SiteMinder Web Agent to protect CA Identity Manager resources.
	3. Install the plug-in the Web Server uses to forward requests to the application server.

✓	Step
	4. Configure the SiteMinder Policy Store for use with CA Identity Manager.
	5. Start the application server and other servers in the installation.
	6. Verify that the plug-in is successfully forwarding requests to the application server.
	7. (Optional) Configure SiteMinder high availability for CA Identity Manager.

How to Install High Availability Provisioning Components

The following table describes the steps involved in installing provisioning components for high availability:

✓	Step
	1. Install primary and alternate Provisioning Servers and provisioning directories for load balancing and failover.
	2. Install several connector servers for load balancing and failover.
	3. Enable clients of the provisioning server to fail over.

How to Uninstall CA Identity Manager

To fully uninstall CA Identity Manager, remove CA Identity Manager software components and clean up the CA Identity Manager-specific configuration in your application server. The following checklist describes the steps to uninstall CA Identity Manager:

✓	Step
	1. Delete CA Identity Manager objects with the Management Console.
	2. (Optional) If you used SiteMinder, remove the CA Identity Manager schema from the policy store or remove the Policy Server. For more information, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide*.

✓	Step
	3. Use the highavailability command to uninstall Provisioning Directories and Provisioning Servers from this location: *Unpacked-Install-Package*\Provisioning\Provisioning Directory\highavailability
	4. Uninstall the CA Identity Manager components.
	5. Remove CA Identity Manager configuration information from the application server.

Index

I

L

M

N

O

P

R

S

T

U

V

W