



Sun Identity Manager Overview



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 820-5819
February 2009

Copyright 2009 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, GlassFish, Javadoc, JavaServer Pages, JSP, JDBC, JDK, JRE, MySQL, Netbeans, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. or its subsidiaries in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc. ORACLE is a registered trademark of Oracle Corporation.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2009 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs brevets américains ou des applications de brevet en attente aux Etats-Unis et dans d'autres pays.

Cette distribution peut comprendre des composants développés par des tierces personnes.

Certains composants de ce produit peuvent être dérivées du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, GlassFish, Javadoc, JavaServer Pages, JSP, JDBC, JDK, JRE, MySQL, Netbeans, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc., ou ses filiales, aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc. ORACLE est une marque dpose registre de Oracle Corporation.

L'interface d'utilisation graphique OPEN LOOK et Sun a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de cette publication et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des Etats-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

Contents

- Preface5**

- 1 Product Overview11**
 - What is Identity Manager? 11
 - How Does Identity Manager Interface With Other IT Systems? 12
 - How Do Users Connect to Identity Manager? 13
 - What is Identity Manager Service Provider? 14
 - Getting to Know Sun's Other Identity Management Products 14
 - What is Sun Java System Directory Server Enterprise Edition? 14
 - What is OpenSSO Enterprise? 15
 - What is Sun Role Manager? 15

- 2 Product Architecture17**
 - Understanding Identity Manager Components 17
 - Understanding the Application Tier 18
 - Understanding the Database Tier 19
 - Understanding the Managed Resource Tier 19
 - Understanding the User Tier 20
 - Understanding the System Separation and Physical Proximity Guidelines 20
 - Understanding SPML and the Web Services System Architecture 21
 - Understanding Identity Manager Service Provider System Architecture 21

- 3 Clustering and High Availability25**
 - Assessing the Need for Availability 25
 - Assessing the Cost of Downtime 25
 - Understanding the Causes of Downtime 27
 - Calculating Return on Investment 27

Understanding the Identity Manager High-Availability Feature Set	27
Making the Repository Highly-Available	28
Making the Application Server Highly Available	29
Making the Gateway Highly Available	29
Understanding the Recommended HA Architecture	30
Understanding the Recommended Service Provider HA Architecture	32
Understanding Failure Scenarios	34
Scenario 1: The No-Workflow Scenario	34
Scenario 2: The Workflow-in-Progress Scenario	35
Scenario 3: The Workflow-in-Abeyance-or-Sleep Scenario	36
Scenario 4: The Work-Item-Edit Scenario	36
Scenario 5: The Scheduled-Tasks-in-Progress Scenario	37
Scenario 6: The Scheduled-Task-in-Abeyance Scenario	38
Scenario 7: The Web-Services-Workflow-Request-Not-Yet-Received-by-Identity Manager Scenario	38
Scenario 8: The Web-Services-Workflow-Request-In-Progress-by-Identity Manager Scenario	39
Frequently Asked Questions Regarding Session Affinity and Session Persistence	40

Preface

Sun Identity Manager 8.1 Overview answers the question *What is Sun™ Identity Manager and how does it work?* The book describes Identity Manager product architecture, as well as information on how to plan a high-availability deployment.

Who Should Use This Book

This guide is for IT professionals who are looking to better understand Sun Identity Manager 8.1 and associated software. It will be of special value to IT professionals who are either in the process of evaluating Identity Manager, or who are in the beginning stages of planning an Identity Manager deployment.

How This Book Is Organized

This guide is organized into the following chapters:

[Chapter 1, “Product Overview,”](#) describes the purpose of Identity Manager and highlights the application's major features.

[Chapter 2, “Product Architecture,”](#) describes the Identity Manager architecture, the Service Provider architecture, and the web services architecture. Guidelines for system separation and physical proximity are also covered.

[Chapter 3, “Clustering and High Availability,”](#) provides guidance on how to implement a high availability / fault tolerant (HA/FT) Identity Manager environment. It will also help you assess the amount of availability that your Identity Manager deployment requires.

Related Books

The Sun Identity Manager 8.1 documentation set includes the following books.

Primary Audience	Title	Description
All Audiences	<i>Sun Identity Manager Overview</i>	Provides an overview of Identity Manager features and functionality. Provides product architecture information and describes how Identity Manager integrates with other Sun products, such as Sun Open SSO Enterprise and Sun Role Manager.
	<i>Sun Identity Manager 8.1 Release Notes</i>	Describes known issues, fixed issues, and late-breaking information not already provided in the Identity Manager documentation set.
System Administrators	<i>Installation Guide</i>	Describes how to install Identity Manager and optional components such as the Sun Identity Manager Gateway and PasswordSync.
	<i>Upgrade Guide</i>	Provides instructions on how to upgrade from an older version of Identity Manager to a newer version.
	<i>System Administrator's Guide</i>	Contains information and instructions to help system administrators manage, tune, and troubleshoot their Identity Manager installation.
Business Administrators	<i>Business Administrator's Guide</i>	Describes how to use Identity Manager provisioning and auditing features. Contains information about the user interfaces, user and account management, reporting, and more.

Primary Audience	Title	Description
System Integrators	<i>Deployment Guide</i>	Describes how to deploy Identity Manager in complex IT environments. Topics covered include working with identity attributes, data loading and synchronization, configuring user actions, applying custom branding, and so on.
	<i>Deployment Reference</i>	Contains information about workflows, forms, views, and rules, as well as the XPRESS language.
	<i>Resources Reference</i>	Provides information about installing, configuring, and using resource adapters.
	<i>Service Provider 8.1 Deployment</i>	Describes how to deploy Sun Identity Manager Service Provider, and how views, forms, and resources differ from the standard Identity Manager product.
	<i>Web Services Guide</i>	Describes how to configure SPML support, which SPML features are supported (and why), and how to extend support in the field.

Documentation Updates

Corrections and updates to this and other Sun Identity Manager publications are posted to the Identity Manager Documentation Updates website:

<http://blogs.sun.com/idmdocupdates/>

An RSS feed reader can be used to periodically check the website and notify you when updates are available. To subscribe, download a feed reader and click a link under Feeds on the right side of the page. Starting with version 8.0, separate feeds are available for each major release.

Related Third-Party Web Site References

Third-party URLs are referenced in this document and provide additional, related information.

Note – Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- [Documentation](http://www.sun.com/documentation/) (<http://www.sun.com/documentation/>)
- [Support](http://www.sun.com/support/) (<http://www.sun.com/support/>)
- [Training](http://www.sun.com/training/) (<http://www.sun.com/training/>)

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. To share your comments, go to <http://docs.sun.com> and click Feedback.

Typographic Conventions

The following table describes the typographic conventions that are used in this book.

TABLE P-1 Typographic Conventions

Typeface	Meaning	Example
AaBbCc123	The names of commands, files, and directories, and onscreen computer output	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. <code>machine_name%</code> you have mail.
AaBbCc123	What you type, contrasted with onscreen computer output	<code>machine_name%</code> su Password:
<i>aabbcc123</i>	Placeholder: replace with a real name or value	The command to remove a file is <i>rm filename</i> .

TABLE P-1 Typographic Conventions (Continued)

Typeface	Meaning	Example
<i>AaBbCc123</i>	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the <i>User's Guide</i> . A <i>cache</i> is a copy that is stored locally. Do <i>not</i> save the file. Note: Some emphasized items appear bold online.

Shell Prompts in Command Examples

The following table shows the default UNIX® system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P-2 Shell Prompts

Shell	Prompt
C shell	machine_name%
C shell for superuser	machine_name#
Bourne shell and Korn shell	\$
Bourne shell and Korn shell for superuser	#

Product Overview

This chapter describes the purpose of Sun™ Identity Manager and highlights the application's major features. It also briefly describes other identity management product offerings from Sun.

The chapter includes the following topics:

- “What is Identity Manager?” on page 11
- “How Does Identity Manager Interface With Other IT Systems?” on page 12
- “How Do Users Connect to Identity Manager?” on page 13
- “What is Identity Manager Service Provider?” on page 14
- “Getting to Know Sun's Other Identity Management Products” on page 14

What is Identity Manager?

Sun Identity Manager makes it possible to automate the process of creating, updating, and deleting user accounts across multiple IT systems. Collectively, this process is known as *provisioning* (that is, creating and updating user accounts) and *deprovisioning* (deleting user accounts).

For example, when an employee joins a company, Identity Manager runs a workflow that retrieves the necessary approvals to grant the employee access. When these approvals are obtained, Identity Manager creates accounts for the employee in the company's human resources system (PeopleSoft), email system (Microsoft Exchange), and enterprise application (SAP). If the employee changes roles in the company, Identity Manager updates the user account and extends access to the necessary resources required in that new role. And when the employee leaves the company, Identity Manager automatically removes the user's accounts to prevent further access.

Identity Manager can also enforce audit policies on an ongoing basis. An *audit policy* specifies what types of access a user may or may not have. For example, in the United States it is a violation of Sarbanes-Oxley (SOX) for the same user to have access to both Accounts Payable and Accounts Receivable systems. This is known as a separation of duties violation. Identity

Manager can conduct audit scanning to check for a variety of these types of violations and, depending on configuration, automatically remove access or send a notification to an administrator when a violation is detected. This process is known as *remediation*.

How Does Identity Manager Interface With Other IT Systems?

In Identity Manager, managed applications and other IT systems are called *resources*. Identity Manager uses either *adapters* or *connectors* to interface with resources.

Adapters and connectors are installed on the Identity Manager server. (Identity Manager does not require special software (called *agents*) to be installed on target resources.) Dozens of Identity Manager adapters and connectors are available, and new ones can be created to communicate with almost any resource using standard protocols or known application programming interfaces (APIs). Identity Manager ships with various adapters and connectors to communicate with many of the most common resources. In addition, templates and skeleton code is available to assist programmers in creating additional adapters and connectors.

Some resources cannot be communicated with directly and require the use of the Sun Identity Manager Gateway. Examples of resources that require the Gateway include Microsoft products, such as Exchange and Windows Active Directory, Novell products, such as eDirectory (formerly Netware Directory Services), and several others. In such cases, Identity Manager communicates directly with the Gateway and the Gateway interfaces with the resource.

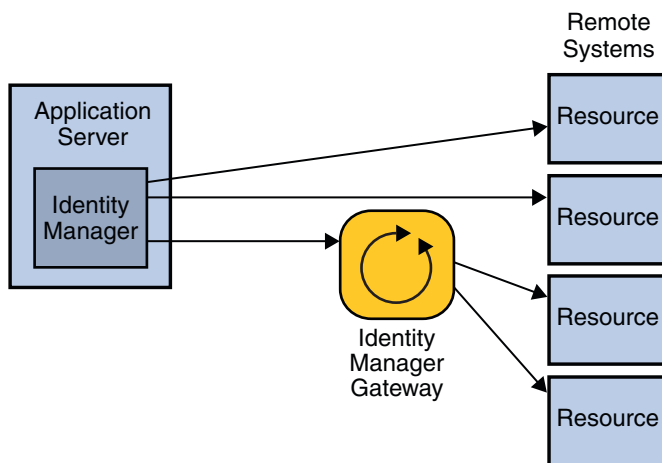


FIGURE 1-1 Identity Manager Interfaces with Some Resources Directly, While Other Resources Require the Identity Manager Gateway

For a list of resources that Identity Manager supports, see [“Supported Resources” in Sun Identity Manager 8.1 Release Notes](#).

How Do Users Connect to Identity Manager?

Identity Manager has a user interface (UI) for administrators, and a separate interface for end users. To use Identity Manager, administrators and end users use a web browser to log on to Identity Manager.

- Administrators use the *administrator interface* to manage users, set up and assign resources, define rights and access levels, establish audit policies, manage compliance, and perform other business administrator and system administrator functions.
- End users use the *end-user interface* to perform a range of self-service tasks, such as changing passwords, setting answers to authentication questions, requesting access to IT systems, and managing delegated assignments.

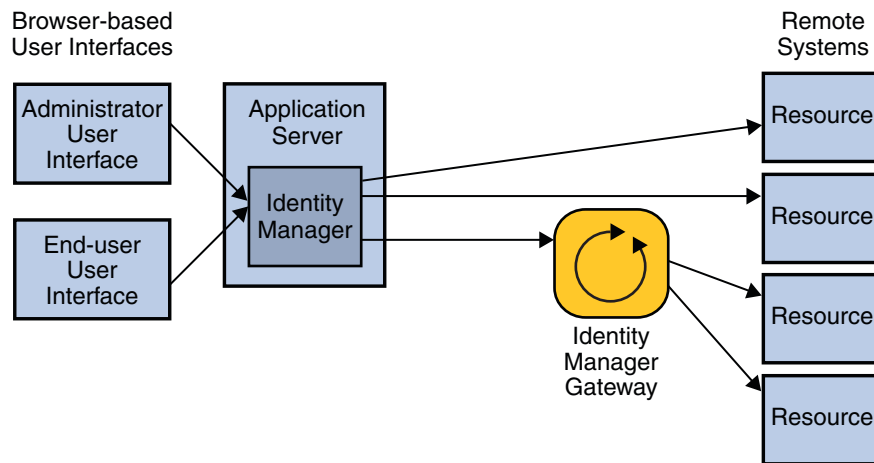


FIGURE 1-2 Users Can Connect to Identity Manager Using the Administrator Interface and the End-User Interface

Companies can also use SPML (Service Provisioning Markup Language) to either create their own user interface, or integrate an existing front-end system with Identity Manager.

Other Identity Manager interfaces include the following:

- The IVR (Interactive Voice Response) telephone interface, which enables end users to perform Identity Manager functions using a telephone
- The Identity Manager IDE (Integrated Development Environment), which is used by software developers to customize Identity Manager
- The Identity Manager console, which is a command-line interface available to administrators

What is Identity Manager Service Provider?

Identity Manager Service Provider is a highly scalable, extranet-focused identity management feature that is capable of provisioning and maintaining millions of end user accounts that are stored on an LDAP directory server. The Service Provider feature can also manage thousands of administrator accounts and synchronize LDAP account data with other resources.

The Service Provider feature uses a subset of the features and functionality available in Identity Manager. For example, auditing functionality is not available because it is less useful in an extranet environment.

For a detailed accounting of the differences between standard Identity Manager and the Service Provider feature, see [“Service Provider Features” in *Sun Identity Manager Service Provider 8.1 Deployment*](#).

Once available as a separate add-on product, Service Provider is now part of Identity Manager. Taking advantage of Service Provider functionality, however, requires special planning.

- For information on how the Identity Manager Service Provider system architecture, see [“Understanding Identity Manager Service Provider System Architecture” on page 21](#).
- For information on planning a highly-available Identity Manager Service Provider architecture, see [“Understanding the Recommended Service Provider HA Architecture” on page 32](#).
- For information on deploying Identity Manager to take advantage of the Service Provider feature, see [Sun Identity Manager Service Provider 8.1 Deployment](#).

Getting to Know Sun's Other Identity Management Products

In addition to Identity Manager, Sun's other identity management solutions include Sun Java™ System Directory Server Enterprise Edition, Sun OpenSSO Enterprise, and Sun Role Manager. These products complement Identity Manager, and, in the case of Role Manager, can extend the capabilities of Identity Manager.

What is Sun Java System Directory Server Enterprise Edition?

Sun Java System Directory Server Enterprise Edition is a scalable, high-performance LDAP data store for identity information. Directory Server Enterprise Edition provides core directory services, as well as other complementary data services. Competing directory service offerings include Active Directory from Microsoft and eDirectory from Novell.

What is OpenSSO Enterprise?

Sun OpenSSO Enterprise (formerly Sun Java System Access Manager and Sun Java System Federation Manager) centralizes and enforces a comprehensive security policy for internal and external applications and web services. It provides secure and centralized access control and single sign-on (SSO) functionality. And it allows for federated identity management, which makes it possible to share applications with companies that have different directory services, security, and authentication technologies. Federated partners trust each other to authenticate their respective users and vouch for their right to access services.

What is Sun Role Manager?

Sun Role Manager (formerly Vaau RBACx) simplifies access control compliance by managing access based on a user's roles within a company and not on an individual, user-by-user basis. By creating roles based on usage and enterprise policies, companies can gain greater visibility into access and manage it in a more efficient, secure, and compliant manner.

Product Architecture

This chapter provides an overview of the Sun™ Identity Manager product architecture.

It includes the following topics:

- [“Understanding Identity Manager Components” on page 17](#)
- [“Understanding the System Separation and Physical Proximity Guidelines” on page 20](#)
- [“Understanding SPML and the Web Services System Architecture” on page 21](#)
- [“Understanding Identity Manager Service Provider System Architecture” on page 21](#)

Understanding Identity Manager Components

Identity Manager is a Java 2 Platform, Enterprise Edition (J2EE™ platform) web application. The J2EE platform consists of a set of industry-standard services, APIs, and protocols that provide the functionality for developing multitiered, web-based, enterprise applications.

The Identity Manager system architecture is distributed across four logical tiers:

- The user tier
- The application tier
- The database tier
- The managed resources tier

Each tier is discussed in the following sections, starting with the application tier.

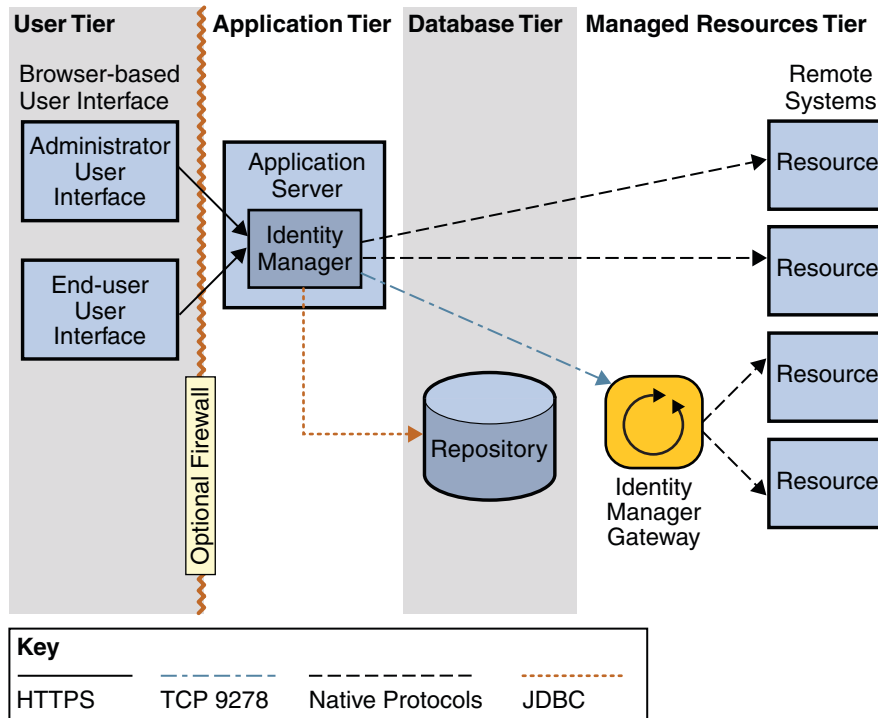


FIGURE 2-1 Identity Manager System Architecture

Understanding the Application Tier

Identity Manager (also known as the Identity Manager server) is installed in a J2EE web container inside an application server. Identity Manager server consists of JSP™ files, HTML, images, and Java™ classes. Adapters and connectors, which interface with other IT systems (also known as *resources*), are also located in Identity Manager on the application server.

Note – See “[Application Servers](#)” in *Sun Identity Manager 8.1 Release Notes* for a list of supported application servers.

Because Identity Manager is a web application, the user interface resides on the application server and pages are served to the user tier on a request-by-request basis.

Installing Identity Manager on the application server is straightforward: A graphical, wizard-based installer is provided, and, on UNIX® systems, a command-line installer is also available. The application server must have a bundled or installed Java Development Kit (JDK™) to run the Java classes that perform actions within Identity Manager.

Understanding the Database Tier

Identity Manager stores all of its provisioning and state information in the Identity Manager *repository*. The repository is comprised of tables that store all the configuration data about Identity Manager. It is a single point for Identity Manager to look up data and lock objects. The repository also contains an audit log, which is a history of actions taken in Identity Manager. Identity Manager data is stored as XML. The repository can reside in local files or a relational database, although in production, a relational database is required.

Note – See “[Repository Database Servers](#)” in *Sun Identity Manager 8.1 Release Notes* for a list of supported database servers.

Note that, beyond a minimal amount of identity information about individual users, user data is not kept in Identity Manager. Instead, only those attributes that are needed to identify and differentiate users within Identity Manager (for example, *name* and *email address*) are saved in the repository.

Identity Manager can connect to the repository over a direct JDBC connection, or it can use data source functionality made available by your application server.

The Identity Manager Service Provider feature requires an additional LDAP repository for storing user information. See “[Understanding Identity Manager Service Provider System Architecture](#)” on page 21 for details.

Understanding the Managed Resource Tier

The managed resource tier consists of the applications and IT systems to which you provision and deprovision user accounts. It includes the Identity Manager Gateway, which is a helper application that allows Identity Manager to interact with certain resources.

Adapters and connectors provide user management functions, including creating, updating, deleting, and reading user accounts, and performing password change management functionality. Adapters and connectors can also extract account information from a remote system.

Note – In most cases, Identity Manager manages user data on the remote system and does not maintain it in its own data store.

Some common resources that require the use of the Sun Identity Manager Gateway include Microsoft Exchange, Windows Active Directory, Novell eDirectory (formerly Netware Directory Services), Lotus Domino, and several others. (See “[Sun Identity Manager Gateway](#)” in

[Sun Identity Manager 8.1 Release Notes](#) for a complete list.) The Gateway installs as a service in Windows and communicates with Identity Manager using TCP port 9278. Communication is initiated from Identity Manager using a proprietary encrypted protocol. The Gateway then interfaces with managed resources using the resources native protocols.

From an installation perspective, there are two type of adapters and connectors: *Identity Manager adapters and connectors* and *custom adapters and connectors*. Identity Manager adapters and connectors are pre-installed in Identity Manager. Custom adapters and connectors, however, need to be copied to a designated directory in the Identity Manager installation directory located on the application server.

Custom adapters are easy to create using the Identity Manager *Resource Extension Facility (REF)* kit. The REF kit provides the API and a number of template adapters that companies can use to jump start the development process. Simple resource functionality can be achieved by implementing only eight Java methods.

Understanding the User Tier

The user tier consists of administrators and end users who interact with Identity Manager through one of the user interfaces. The main user interface for the product is a web browser, which communicates with Identity Manager over HTTPS. The two browser-based UIs, the *administrator user interface* and the *end-user interface*, primarily consist of HTML pages, although some features may use Java applets.

For clarity, only the administrator user interface and the end-user user interface are shown in figure [Figure 2–1](#). Other user interfaces, however, are also located in the user tier. These include the IVR telephone interface, the Identity Manager IDE, the SPML web services interface, and the Identity Manager console.

Understanding the System Separation and Physical Proximity Guidelines

This section contains basic guidelines on what Identity Manager components should run on what servers. It also contains recommendations on which components should be physically sited near one another in order to minimize performance issues that could arise due to latency and network congestion.

Note – Only basic guidelines are provided. For information on designing a high-availability Identity Manager architecture, see [Chapter 3, “Clustering and High Availability.”](#)

In a development environment, the application server and database can reside on the same machine. In testing and production environments, however, each Identity Manager instance should be installed on its own dedicated server. The relational database also requires a dedicated server.

The Identity Manager Gateway, if required, must be installed on one or more Windows machines. The Gateway is a lightweight component and does not require a dedicated server. All Windows domains managed by a Gateway must be part of the same forest. Managing domains across forest boundaries is unsupported. If you have multiple forests, install at least one Gateway in each forest. In production the Gateway must be made highly available. See [“Making the Gateway Highly Available” on page 29](#) for details.

In a production environment, the highest amount of network traffic occurs between the database and application servers. These two environments must be on the same LAN with the shortest network hop possible. Gateway instances, as well as managed resources, do not need to be on the same network as Identity Manager.

If Identity Manager will be used for external users in a Service Provider configuration, a set of web servers should be setup in a DMZ. See [“Understanding the Recommended Service Provider HA Architecture” on page 32](#) for details.

Understanding SPML and the Web Services System Architecture

Service Provisioning Markup Language (SPML) and Identity Manager Web Services can be used to implement a custom front-end for Identity Manager. Identity Manager sends and receives SPML messages and responses using the HTTPS protocol.

For more information about SPML and Web Services, see [Sun Identity Manager 8.1 Web Services](#).

Understanding Identity Manager Service Provider System Architecture

If the Identity Manager Service Provider feature is implemented, a fifth tier is required. This tier is called the Web tier and it consists of one or more web servers located in a DMZ. No Identity Manager components are installed in the web tier. Instead, the web servers in the DMZ support one or more application servers in the application tier by responding to web page requests. Adding one or more web servers to the web tier provides enhanced scalability, and placing the web servers in a DMZ provides better network security.

The Service Provider feature also requires an LDAP repository. This repository resides in the database tier. Because the LDAP repository can be a managed resource, the LDAP server can be understood as residing in the managed resource tier, as well.

Note – In a service-provider-only implementation, an Identity Manager repository is recommended in addition to the LDAP repository, but it is not required. If an Identity Manager repository is not deployed, some functionality such as certain reporting capabilities will not be available.

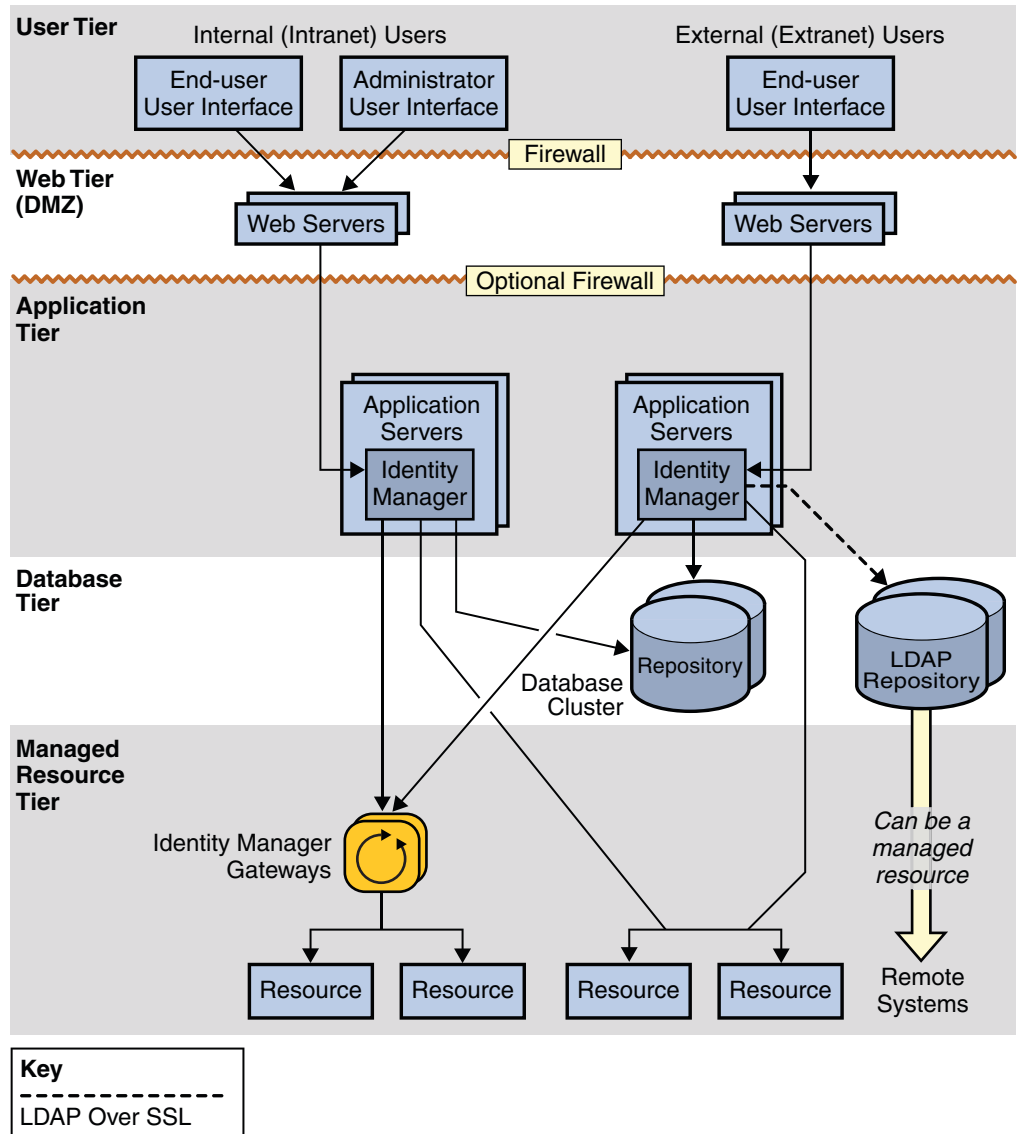


FIGURE 2-2 Identity Manager Service Provider System Architecture

Clustering and High Availability

This chapter provides guidance on how to implement a high availability / fault tolerant (HA/FT) Identity Manager environment.

Note – Please consult your web server, application server, and database provider's documentation for best practices on ensuring a highly available deployment with each technology. This guide is not a substitute for the vendor-specific recommendations for web servers.

- [“Assessing the Need for Availability” on page 25](#)
- [“Understanding the Identity Manager High-Availability Feature Set” on page 27](#)
- [“Understanding the Recommended HA Architecture” on page 30](#)
- [“Understanding the Recommended Service Provider HA Architecture” on page 32](#)
- [“Understanding Failure Scenarios” on page 34](#)
- [“Frequently Asked Questions Regarding Session Affinity and Session Persistence” on page 40](#)

Assessing the Need for Availability

This section describes how to assess the amount of availability that your specific deployment requires.

Assessing the Cost of Downtime

Because Identity Manager is not in the transaction path between general users and the systems and applications that they already have access to, Identity Manager downtime is not the nightmare that you might imagine. If Identity Manager is unavailable, end users are still able to access resources through their provisioned accounts.

The main cost of Identity Manager downtime is lost productivity. If Identity Manager is down, end users cannot use Identity Manager to gain access to systems that they are either locked out of or not provisioned to.

To calculate the cost of downtime, the first number that is needed is the average cost of lost productivity due to end users being unable to access computing resources within the enterprise. In our assessment, this number is called *productivity per person hour*.

The other major number that needs to be determined is the percentage of end users within the user population who need to use Identity Manager at any given time. This population usually includes new hires who need to be provisioned, and end users who have forgotten their password if password management is a part of the deployment.

Consider the following hypothetical situation:

Total number of employees	20,000
Number of password resets in a day	130
Number of new hires in a day	30
Number of hours in a work day	8

For this particular situation you can calculate the following:

- The number of employees needing Identity Manager at any given hour = $(130 + 30) / 8 = 20$
- The percentage of employees needing Identity Manager at any given hour = $20 / 20,000 = .1\%$ or 1 in 1000

Using these numbers you can then estimate the cost of an Identity Manager outage:

Productivity per person hour	\$100	
Loss in productivity	.5	(50% decrease in productivity due to inability to access system)
Number of people affected	20	
Subtotal	\$1,000	
Duration of outage	2 hours	
Total immediate loss	\$2,000	

This example shows that even though the number of users being managed by Identity Manager is high, the number of users needing Identity Manager to gain access to systems at any given time is usually low.

Another point to consider is that the time it takes to bring a system like Identity Manager back online is usually less than the time it takes to execute the manual provisioning processes that Identity Manager is automating. So while Identity Manager downtime exacts a cost, it is usually less than the cost of using manual processes to give users access to resources.

Understanding the Causes of Downtime

When planning for an Identity Manager highly-available deployment, it is worthwhile to consider the causes of downtime.

These causes include the following:

- Operator error
- Hardware failure
- Software failure
- Planned down time (Upgrades to hardware and software)
- Poor performance (Perceived downtime)

Calculating Return on Investment

Identity Manager automates processes and reduces lost productivity. The return on investing in a highly-available Identity Manager architecture is realized by minimizing downtime and averting lost productivity.

You can use the cost of downtime to determine the amount of availability that is ultimately needed for Identity Manager. In general, a moderate investment in making Identity Manager highly-available is worthwhile.

When calculating the cost of your investment, remember that purchasing HA/FT hardware and software is only one part of implementing an available solution. Having a knowledgeable staff to keep it up and running is another cost.

Understanding the Identity Manager High-Availability Feature Set

Identity Manager is designed to leverage HA infrastructure if it is available. For example, Identity Manager does not require an application server cluster to achieve high availability, but it can utilize a cluster if it exists.

The following diagram shows the major Identity Manager components deployed in a non-redundant architecture. The sections that follow will describe how the Identity Manager repository, application server, and gateway can be made highly-available.

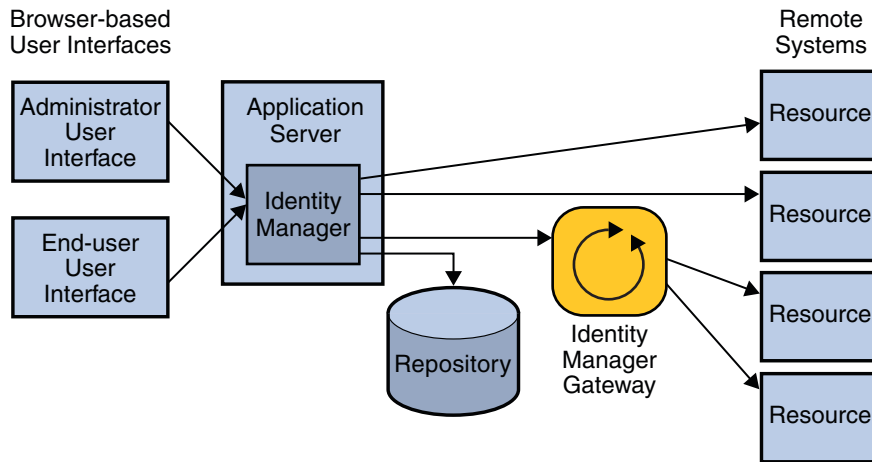


FIGURE 3-1 Identity Manager Standard System Architecture

Note – See [“Understanding the System Separation and Physical Proximity Guidelines” on page 20](#) for information on which components should be physically sited near one another in order to minimize performance issues that could arise due to latency and network congestion.

Making the Repository Highly-Available

Identity Manager stores all of its provisioning and state information in the Identity Manager repository.

The availability of the database instance storing the Identity Manager repository is the most critical piece to achieving a highly available Identity Manager deployment. The repository is the representation of the entire Identity Manager installation and the data within it must be protected as with other important database applications. At minimum, regular backups must be performed.

Note – Do not host the Identity Manager repository on a virtual platform such as a VMware virtual machine because performance (transactions per second) will be hindered significantly.

There can only be one image of the repository. It is not possible to have two separate databases for Identity Manager and attempt to synchronize them nightly. Sun recommends using your database's clustering or mirroring capabilities to provide fault tolerance.

Making the Application Server Highly Available

Identity Manager can run within an application server cluster and take advantage of the added availability and load balancing that a cluster provides. Identity Manager does not use any J2EE features that require clustering, however.

Identity Manager uses the HTTP Session object that is available through the Servlet API. This session object tracks a user's visit as the user logs in and performs actions. In a cluster, you can optionally have multiple nodes handle a user's requests during a given session. This is usually not recommended, however, and most installations are configured to send a user's entire request for a given session to the same server.

It is possible to add additional availability and capacity to the application server running Identity Manager even if you do not set up a cluster. This is achieved by installing multiple application servers with Identity Manager, connecting them to the same repository, and putting a load balancer with *session affinity* in front of all the application servers.

Note – For more information on session affinity, see [“Frequently Asked Questions Regarding Session Affinity and Session Persistence” on page 40.](#)

Identity Manager runs certain tasks in the background—for example, scheduled reconciliation tasks. These tasks are stored in the database and can be picked up by any Identity Manager server to run. Identity Manager uses the database to ensure that these tasks are always run to completion, even if it has to fail over to another node.

Configuring Active Sync Clustering on Application Server Nodes

The `sources.hosts` setting in the `Waveset.properties` file controls which hosts in a multi instance environment are used for executing Active Sync requests. This setting provides a list of hosts that source adapters can run on. Setting this to `localhost` or `null` will allow source adapters to execute on any host in the web farm. (This is the default behavior.) By listing one or more hosts, you can restrict execution to that list. If you have inbound updates from another system that go to a particular host, use the `sources.hosts` setting to record the host names.

In addition, you can define a property named `sources.resourceName.hosts`, which controls where the resource's Active Sync task will run. Replace `resourceName` with the name of the resource object you wish to specify.

Making the Gateway Highly Available

Identity Manager requires a lightweight gateway to manage resources that cannot be directly accessed from the server. These include systems that require client-side API calls that are platform specific. For example, if Identity Manager is running on a UNIX-based application

server, the ability to make NTLM or ADSI calls to managed NT or Active Directory domains is not possible. Because Identity Manager requires a gateway to manage these resources, it is important to ensure that the Identity Manager Gateway is made highly available.

To prevent the Gateway from becoming a single point of failure, Sun recommends having multiple machines running a Gateway instance. A network routing device should be configured to provide failover if the main Gateway instance dies. The failover device should be setup for sticky sessions and use a simple round robin scheme. Do not place the Gateways behind a device that load balances! This is not a supported configuration and will cause certain Identity Manager functions to fail.

All Windows domains managed by a Gateway must be part of the same forest. Managing domains across forest boundaries is unsupported. If you have multiple forests, install at least one Gateway in each forest.

Win32 monitoring tools can be configured to watch the gateway . exe process on the Win32 host. In the event that gateway . exe fails, the process can be automatically restarted.

Understanding the Recommended HA Architecture

The following diagram shows the Identity Manager architecture Sun recommends if there is no existing web application infrastructure.

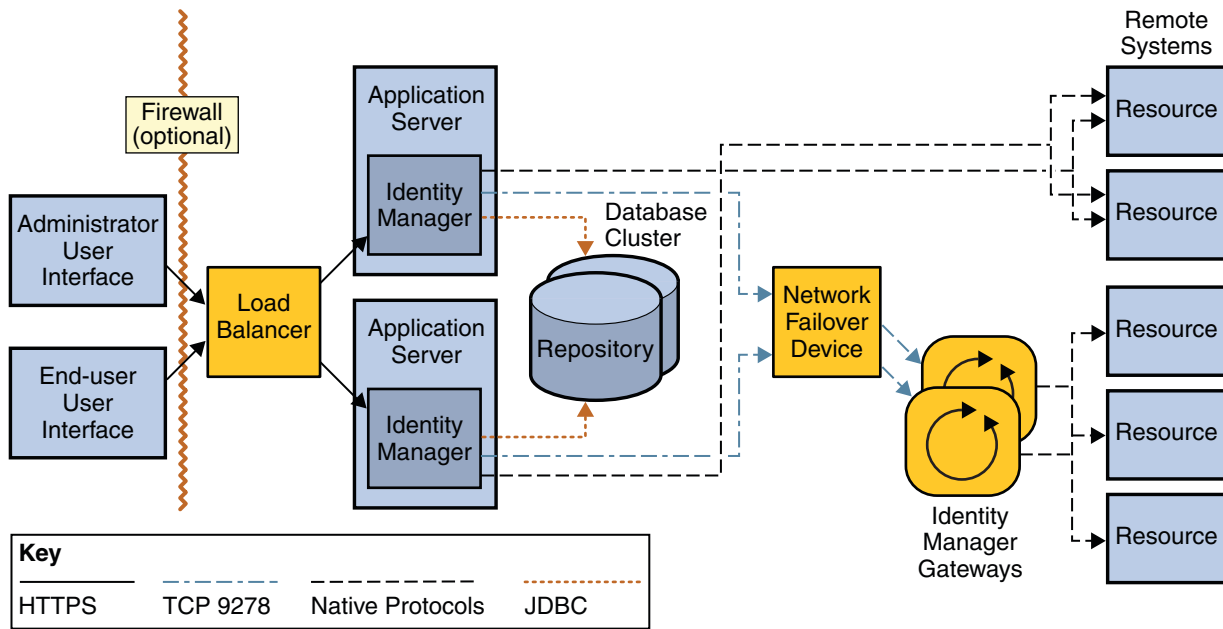


FIGURE 3-2 Identity Manager High-Availability Architecture

In an actual deployment, existing redundant application server infrastructure should be utilized to the extent possible. The value of this architecture is that it only uses load balancers for achieving redundancy at the application server. Load balancers with session affinity detect failed application server instances and failover to active instances. Load balancers are also used to provide horizontal scaling in the web environment by spreading the user requests across a cluster of servers.

Though this is a straightforward architecture, the uptime characteristics are comparable to more complex deployments. Because of its simplicity, there are fewer pieces of software to maintain and monitor or fewer pieces that could fail. Because human error is the number one cause of downtime, a relatively simple solution may achieve better uptime characteristics than something more complex. There are no universal right answers. The point is to understand all of the causes of downtime and choose the architecture that will result in the best availability for the investment.

Note – It would be impossible to describe all of the different HA architectures that are possible with a web application like Identity Manager.

Because Identity Manager can be deployed in a variety of possible combinations, it may be most economical to identify existing infrastructure and utilize as much of it as possible when deploying Identity Manager.

Understanding the Recommended Service Provider HA Architecture

If Identity Manager Service Provider functionality is to be utilized, Sun recommends adding a web tier between the user tier and the application tier. The web tier consists of one or more web servers that reside in a demilitarized zone (DMZ) that is separated by a firewall from the application tier.

An LDAP repository is required if Service Provider functionality is to be utilized. If Identity Manager will only be supporting extranet clients, a standard Identity Manager repository is recommended, but not required. Otherwise, if Identity Manager will be supporting both intranet and extranet users, an LDAP repository and a standard Identity Manager repository is required.

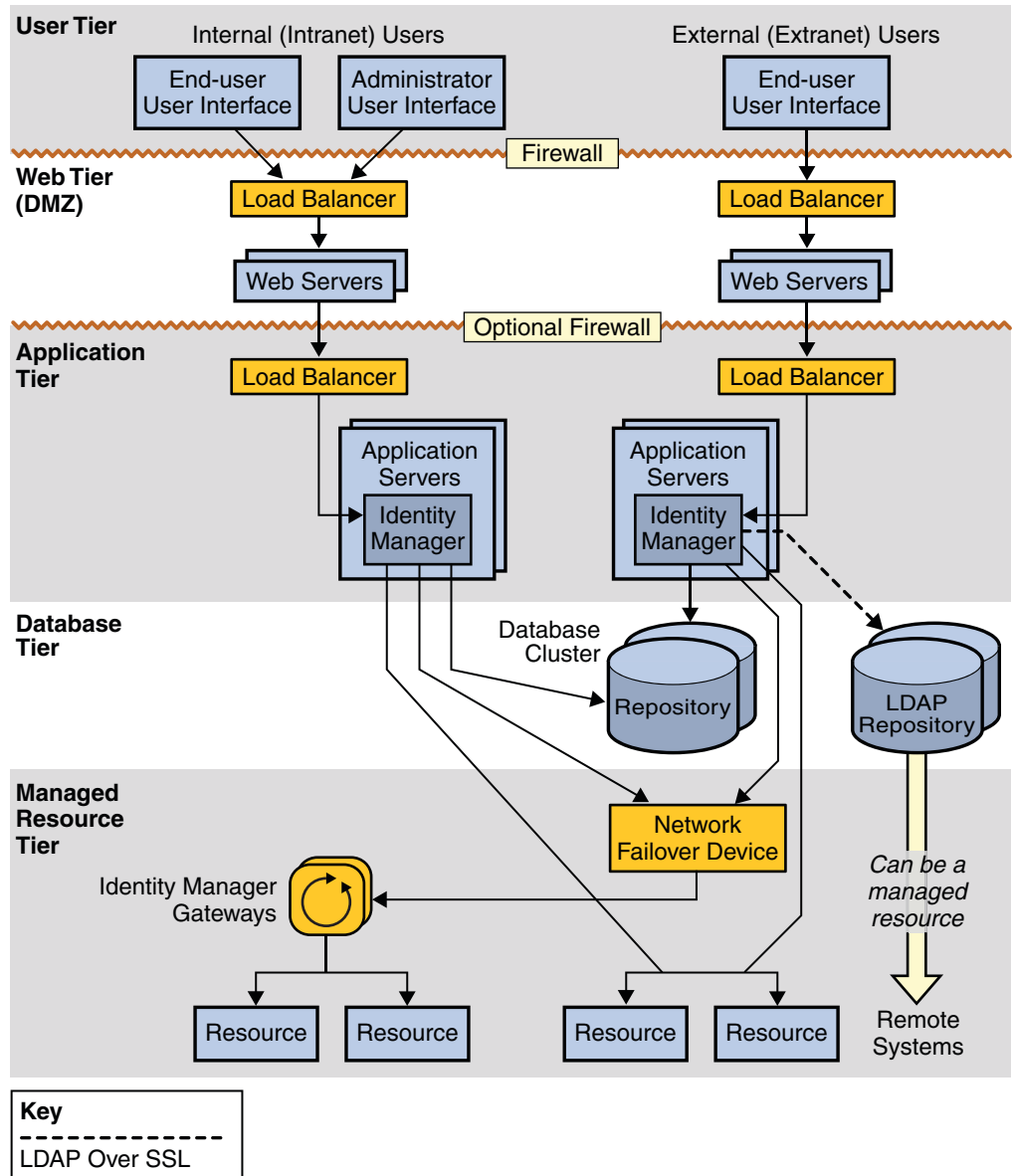


FIGURE 3-3 Identity Manager Service Provider High-Availability Architecture

Understanding Failure Scenarios

This section lists eight failure scenarios and compares two deployments, one with session persistence, and one without.

- The deployment *with* session persistence has session affinity across a load balancer. The deployment has multiple instances in a cluster which have some form of session persistence turned on such that session changes are written to a physically distinct repository node.
- The deployment *without* session persistence has session affinity across a load balancer, and has multiple instances that are not part of a cluster.

Scenario 1: The No-Workflow Scenario

Scenario Description

The end user or the administrator is editing a form that is not a part of a workflow. The instance on which the user has an established session goes down.

Without Session Persistence

User experience: A nontransparent failover. Upon submitting the form, the user is returned to the login page.

Recovery steps: The user reenters his or her user name and password. Identity Manager then processes the form and presents the results as the next page immediately following the login.

With Session Persistence

User experience: The user's form is submitted and the results are returned without the user being logged off and required to log in again.

Recovery steps: No user action is needed.

Other Scenario Examples

- An end user has logged in and retrieved search results for users or other repository objects when the instance goes down.
- An administrator is about to submit a “reset password” or “edit user” request using the Administrator interface when the instance goes down.

Scenario 2: The Workflow-in-Progress Scenario

Scenario Description

The end user or the administrator has submitted a form that triggered a workflow. The instance on which the workflow is executing and on which the user's session exists is generally going to be the same except in case of some scheduled tasks where they can be different. This instances goes down while the workflow is in progress.

Without Session Persistence

User experience: A nontransparent failover. The form submit returns the user to the login page. The workflow task instance being executed should be in the repository, but because the node of execution is down, the workflow status will be “terminated.”

Recovery steps: The workflow has to be submitted again. This has to be done by going back to the same form and reentering the same information that was used to trigger the workflow before the node failed.

The submission of the same request data may work in some cases, but not all. If the workflow provisions to more than one resource during its execution and some of these resources were provisioned before the failure, the workflow resubmission from the user would have to account for the “already provisioned to” resources. Note that the terminated workflow sticks around in the repository until the `resultLimit` expires on the `TaskInstance` object.

With Session Persistence

User experience: A nontransparent failover. The user does not get logged out because his session is persisted and reestablished in the new instance. The form submit, however, will probably result in an error because the workflow will be terminated. This failover is nontransparent because recovery actions are needed.

Recovery steps: Same as in the Without Session Persistence mode. The user has to resubmit the request that triggered off the previous workflow with the same or modified parameters.

Other Scenario Examples

- An end user has just submitted a self-registration request to create an Identity Manager account and the instance goes down.
- An administrator has just submitted a “password reset” request that is in progress and the instances goes down.

Scenario 3: The Workflow-in-Abeyance-or-Sleep Scenario

Scenario Description

This scenario covers situations where the workflow has started, but is waiting for a manual action by an approver.

Without Session Persistence

User experience: Transparent failover with respect to the approver provided that the approver has not yet logged in. After the node failure, when the approver does log in, the approver will still see the approval request in his or her inbox, even though the request was triggered from a node that is no longer up.

Recovery steps: No user action is needed.

With Session Persistence

User experience: Same as in the Without Session Persistence mode.

Recovery steps: Same as in the Without Session Persistence mode.

Other Scenario Examples

- The workflow is in a sleep state, for example a manual action that sleeps until a sunrise or sunset date for an employee.
- An administrator submitted a user creation request that is waiting on an approver to log in and approve the request. The node from which the request was sent failed before the approver approved the request.

Scenario 4: The Work-Item-Edit Scenario

Scenario Description

This scenario includes cases where a user is editing a work item and the node upon which the user has a session goes down before the work item can be submitted.

Without Session Persistence

User experience: A nontransparent failover. When the work item edit form is submitted, the user is logged off and returned to the login page.

Recovery steps: Upon resubmitting login credentials, the user's work item is marked completed and the workflow can resume from that point. The workflow should be picked up by the new mode for execution from the point where the user's manual action is marked completed.

With Session Persistence

User experience: When the work item edit form is submitted, the user sees the effect of his submission—for example, either the next form in the custom workflow if there is one, or a success message.

Recovery steps: No user action needed.

Other Scenario Examples

- An end user is filling out a form associated with a manual action in a custom workflow, for example requesting access to specific resources. Before the user can submit the request, the node the user has a session on dies.
- An administrator has logged in to Identity Manager and has opened up an approval request for editing. Before the request can be submitted, the node the administrator has a session on fails.

Scenario 5: The Scheduled-Tasks-in-Progress Scenario

Scenario Description

These scenarios cover cases where a node failure occurs while a reconciliation is in progress or while a report is executing.

Without Session Persistence

User experience: The scheduled task terminates in process.

Recovery steps: The scheduled task that was in progress has to be restarted. The task will start from the beginning. (The task will not restart from the point of failure.) This is akin to creating and starting a new task.

With Session Persistence

User experience: Same as in the Without Session Persistence mode.

Recovery steps: Same as in the Without Session Persistence mode.

Other Scenario Examples

- An Active Sync adapter is configured to run on the failed node.

Scenario 6: The Scheduled-Task-in-Abeyance Scenario

Scenario Description

These scenarios cover cases where a user's custom workflow has scheduled a task for execution at a later date on a specific node. Before the scheduled date is reached, the node that the task was scheduled on fails.

Without Session Persistence

User experience: The failover is transparent with respect to the recovery actions required to ensure that this task executes at its scheduled time.

Recovery steps: The scheduled task is taken up by any live node when the scheduled execution time arrives.

With Session Persistence

User experience: Same as in the Without Session Persistence mode.

Recovery steps: Same as in the Without Session Persistence mode.

Other Scenario Examples

- In the process of a user's account creation, the Deferred Task Scanner is used to implement enabling an account on a sunrise date or to implement disabling the account on a sunset date. Before the sunrise or sunset date arrives, the node that the task was scheduled on fails.
- A report is scheduled to run at a future time, or a reconciliation is scheduled to run at a specific time and, before the time is reached, the node the task was scheduled on fails.

Scenario 7: The Web-Services-Workflow-Request-Not-Yet-Received-by-Identity Manager Scenario

Scenario Description

These scenarios cover those cases where the Identity Manager GUI is not used to launch provisioning. Instead, the user interface is provided by an application that internally calls to Identity Manager using the SPML or other custom web service interface. Here the user session related to the user going through the UI is managed by way of the calling application. For Identity Manager, the requests are all launched as the “soapadmin” subject.

In such a use case, this failure scenario covers the case where the request by way of the Identity Manager endpoint has not been received yet and the targeted node fails.

Without Session Persistence

User experience: A transparent failover. The SOAP administrator's credentials are passed in for each SOAP request, either over the wire or within Identity Manager through a `Waveset.properties` setting. As long as the node which was to receive this SOAP request has not received the request before going down, the failover is transparent with or without session persistence.

Recovery steps: No action needed. The SOAP request is sent to a live node that executes it.

With Session Persistence

User experience: Same as in the Without Session Persistence mode.

Recovery steps: Same as in the Without Session Persistence mode.

Scenario 8: The Web-Services-Workflow-Request-In-Progress-by-Identity Manager Scenario

Scenario Description

This scenario is similar to scenario seven. The only difference is that the workflow is in progress when the node fails, or the node has received the SOAP request when the node fails.

Without Session Persistence

User experience: This scenario is similar to scenario two (workflow in progress). The workflow is marked terminated and the user sees an error as a result of the SOAP request.

Recovery steps: The user has to resubmit the form with similar or modified parameters (based on where the failure occurs in the workflow) using the user interface in the third-party application.

With Session Persistence

User experience: Same as in the Without Session Persistence mode.

Recovery steps: Same as in the Without Session Persistence mode.

Frequently Asked Questions Regarding Session Affinity and Session Persistence

Should you have session affinity enabled when scaling application servers horizontally?

Yes.

Should you have session persistence when scaling application servers horizontally?

Unless your business requirements place a very high emphasis on having transparent failover in the limited situations where session persistence can make a difference, Sun recommends against using session persistence. Session persistence has its own performance overhead and, unless transparent failovers are absolutely mandated by your business requirements, leave session persistence turned off.

If you study the failure scenarios documented in [“Understanding Failure Scenarios” on page 34](#), in six of the eight scenarios there is no difference in the end-user experience or required recovery actions, regardless of whether session persistence is enabled. Only in scenarios one and four are there any difference between the session-persistence scenarios as opposed to the no-session-persistence scenarios.

In these two scenarios, session persistence can provide some failover transparency, but session persistence hurts performance. Based on the size of the session objects, the repository being used for session persistence, and the optimization of your specific application server's session management code, the performance overhead can range from 10 percent to 20 percent or even higher.

Should you have multiple application server instances in a cluster when scaling horizontally?

Multiple application server instances are not absolutely needed unless you want session persistence. Fail-over without session persistence can be achieved even if all application server nodes are not in a cluster.