HP Server Automation

for the HP-UX, IBM AIX, Red Hat Enterprise Linux, Solaris, SUSE Linux Enterprise Server, VMware, and Windows® operating systems

Software Version: 9.0

Overview and Architecture Guide

Document Release Date: June 2010
Software Release Date: June 2010

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

Trademark Notices

Intel® and Itanium® are trademarks of Intel Corporation in the U.S. and other countries.

Java™ is a U.S. trademark of Sun Microsystems, Inc.

Microsoft®, Windows®, Windows® XP are U.S. registered trademarks of Microsoft Corporation.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Adobe® is a trademark of Adobe Systems Incorporated.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Document Changes

Chapter	Date	Version	Changes
	June 2010	9.0	Document Created

Support

Visit the HP Software Support Online web site at:

www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

Contents

1 Introduction to HP Server Automation

Overview of HP Server Automation (SA)

If you are new to HP Server Automation (SA), you should read this section. If you are an experienced SA user, you should read SA Architecture and Features In-Depth on page 7 to see what has changed in this release.

SA automates critical areas of your data center's server management:

- **Server Discovery**

  You can scan your network for servers that are not yet managed by SA. You can then add them to the SA Managed Server Pool. After the servers are under SA management, you can perform management tasks on them, including:

- **Operating system provisioning**

  This SA feature enables you to provision bare metal servers with a pre-configured operating system and bring them into the SA Managed Server Pool, after which you can perform centralized management of the servers using SA.

- **Software provisioning**

  After a server is part of the managed server pool, you can install and configure software applications using templates called Software Policies. A software policy specifies the software to be installed, the configurations to be applied, and the scripts to be run during installation. Software policies allow you to establish a baseline configuration for your servers which you can then enforce using SA's Software Compliance feature. For example, you can install an baseline version of an Apache server on all or a subset of your SA managed servers.

- **Application Configuration**

  You can design application configuration templates and push those configurations to all SA managed servers. For example, if you have an iPlanet Web server, you might want to ensure that its configuration files are standardized on all servers on which it is deployed. Application configuration allows you to do that.

- **Software Compliance**

  SA's Software Policy Compliance Scan determines whether a Managed Server's software configuration is compliant with the specifications in the software policies attached to that server.

- **Audit and Compliance**

  The Audit and Remediation feature allows you to define server configuration policies to help you ensure that your SA managed servers meet your policy standards. When servers are found to be out of compliance — not configured the way you want them to be — you can remediate them (force them into compliance). You can base your compliance policy on a snapshot of a base server that you have configured as you want all servers to be configured.

SA's audit trail data helps you establish strict accountability in your data center environment — an increasingly urgent topic in the age of Sarbanes-Oxley Act, the Gramm-Leach-Bliley Act (GLB Act), and the Health Information Portability and Accountability Act (HIPAA).

- **Operating System Patching**

  SA provides an automated, centralized, and flexible method of applying the required operating system patches for Windows, Linux and Solaris-based managed servers. You can compare required patches against operating system vendor approved lists. You can customize the patching process to omit patches that are incompatible with a server's environment.

- **Reporting**

  SA provides an extensive set of comprehensive and configurable reports that you to present data about the state of your managed servers for various audiences.

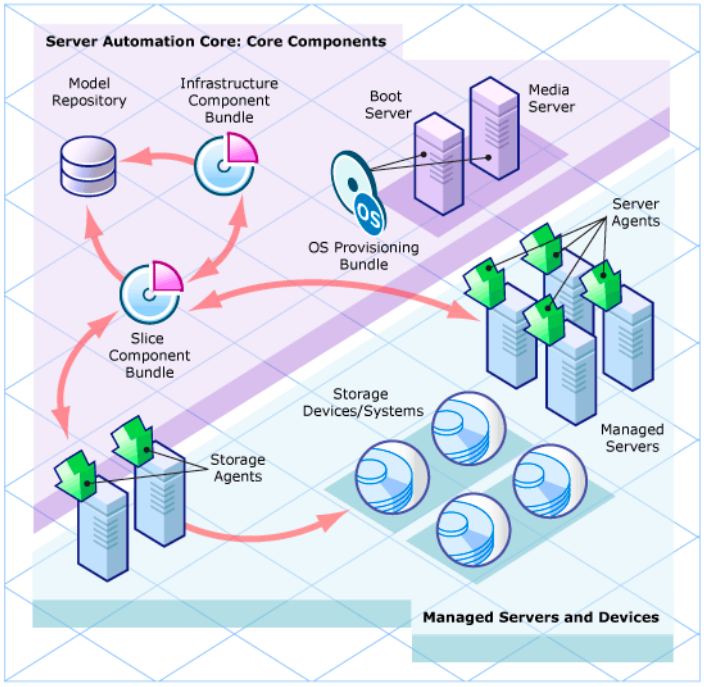
SA allows you to make changes more safely and consistently because you can model and validate changes before you actually commit the changes to a managed server. SA also provides methods to ensure that modifications you plan for your managed servers work on the first time because they have been tested before being applied, thereby reducing downtime.

A Simple SA Installation

SA installs a number of components that provide its server management capabilities. If you have no need to customize your SA installation, you can choose the SA Single-Host Installation. However, if you need to customize your installation, for example distribute SA Core Components to different servers for performance reasons or require a remote database installation, you will need to use the Advanced SA Installation.

Figure 1 shows the simplest SA installation for a single facility. It consists of SA installed on a single host managing the servers on one network.

Figure 1 A Simple SA Installation



The Oracle Database

All SA installations require an Oracle database that is configured specifically for SA and is used to store information about your network, storage devices, managed servers and the operating systems and applications installed on them, and so on. This database is provided as part of the SA installation or you can use an existing Oracle installation that has been configured for use with SA (see "Appendix A: Oracle Setup for the Model Repository" in the *SA Simple/Advanced Installation Guide*

Installation Methods

When you install SA, you can choose one of four installation methods:

- **Single-host Installation**

  This installation allows you to get up-and-running quickly. You must install SA on a single host and use the SA-supplied Oracle database which must be installed on the same host. You need only to know the username and password for the SA administrator. You will use the installation procedure described in the *SA Single-Host Installation Guide*.

- **Simple Installation**

  This installation, while still simple, allows you to distribute SA components to multiple hosts and use a remotely installed Oracle installation. *If you plan to distribute SA Core Components to multiple hosts, you must read* SA Architecture and Features In-Depth on page 7 *and use the Simple Installation procedure described in the SA Simple/Advanced Installation Guide.*

- **Advanced Installation**

  The Advanced installation allows the greatest flexibility in customizing your SA installation including, distributing SA Core Components over multiple hosts, remote Oracle database, SA Satellite installation, Multimaster Mesh installations, custom configurations, etc. *This is a complex installation. If you plan to use this installation method, you must read* SA Architecture and Features In-Depth on page 7 *and use the Advanced Installation procedure described in the SA Simple/Advanced Installation Guide.*

- **Expert Installation**

  This method allows you to change all SA configuration parameters.This method is used only by HP technical support and consulting staff. *Changing the values for certain parameters could irrevocably damage your SA installation. HP recommends that the typical user not use the Expert Installation.*

Accessing SA Features

There are two ways you will typically access SA features:

- **SAS Web Client**

  The web-based user interface to SA through which you manage your servers. Instructions for starting the SAS Web Client in your browser are in the *SA Single-Host Installation Guide*. After you start the SAS Web Client, you can download and install the SA Client Launcher utility from the Power Tools pane.

You use the SAS Web Client to:

— add user accounts and assign permissions

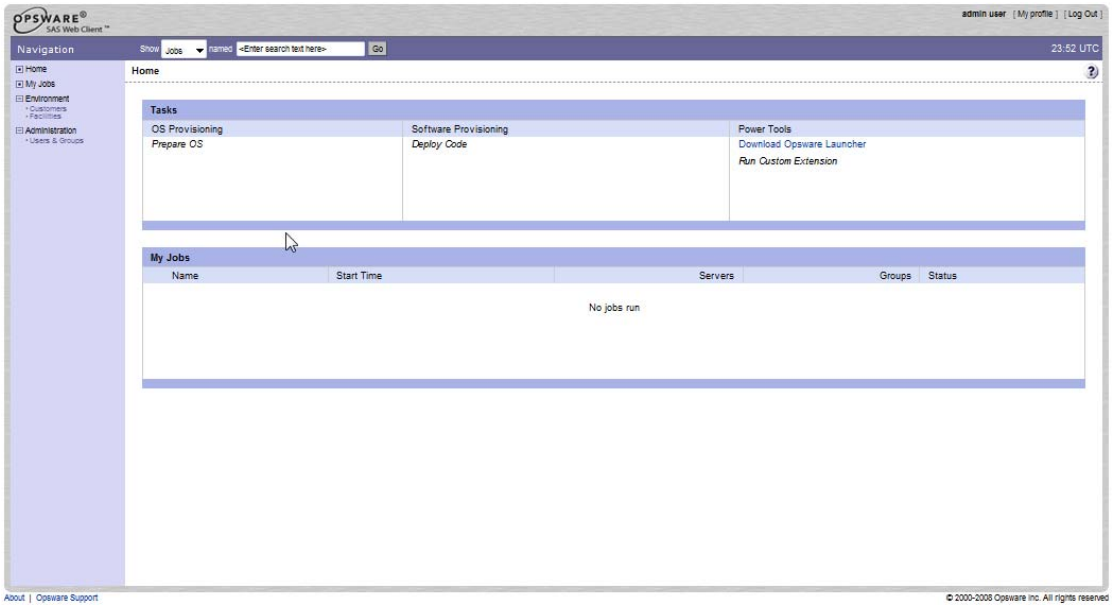
— add customers and facilities

— change SA configuration

— monitor SA and diagnose problems.

Figure 2 shows the SAS Web Client home page.

Figure 2 The SAS Web Client Home Page



- **SA Client**

  A Java Web-Start application that extends the SAS Web Client features and provides access to the following features:

  — Discovery and Agent Deployment

  — Audit and Remediation

  — Compliance View

  — Reports

  — Software Management

  — Patch Management for Windows

  — Patch Management for Unix

  — Patch Management for Solaris

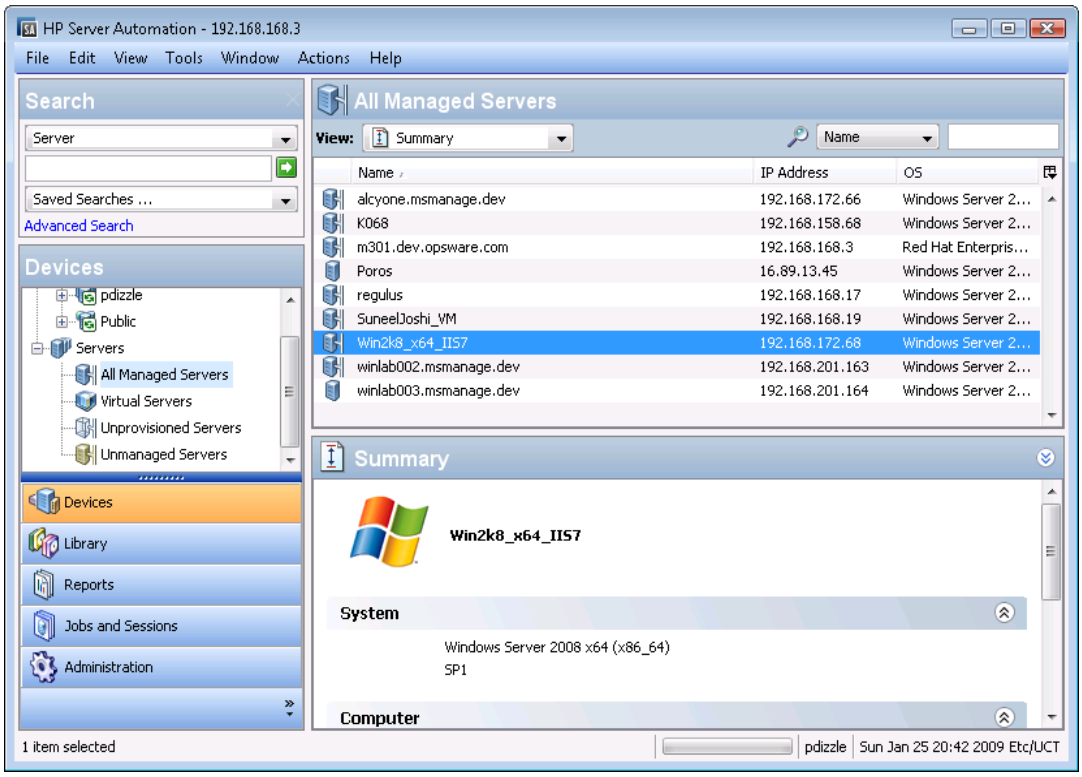
  — Application Configuration Management

  — Virtualization Director

  — Server Automation Visualizer (SAV)

Figure 3 shows the SA Client main screen. You can find detailed information about the SA Client in the *User Guide: Server Automation*.

Figure 3 The SA Client Main Screen



Getting Started with SA after a Simple Installation

After you have installed SA, you must perform the tasks shown below.

1. Launch the SAS Web Client
2. Create an user account with SA administrator privileges.
3. Create an Advanced SA user.
4. Download the SA Client Launcher and install the SA Client.
5. Scan your network for unmanaged servers and bring the them into your SA Managed Server Pool

After completing these tasks, you are ready to begin exploring the various SA features using the information in the next section or in the SA interactive tutorials. Also, see your SA documentation for the specific feature you want to learn how to use.

Sample Use Cases and Tutorials

Chapter 2, SA Feature Overview provides an overview of using SA features. HP also provides interactive tutorials in the *SA Getting Started* guide that walk you through performing certain tasks like installing an operating system on a server, creating a software policy, applying Windows/Solaris patches, and more. It also includes interactive demos.

SA Architecture and Features In-Depth

If you plan to use any installation other than the Simple SA Installation because you need to take advantage of SA's advanced component layout and customization capabilities, see and Chapter 3, SA Features and Utilities In-depth and Chapter 3, Advanced SA Architecture and Options.

2 SA Feature Overview

SA is provides a suite of features that automate data center processes. These features have been designed to replace ad hoc, error-prone, manual processes. For example, by using the OS Provisioning feature, users can set standards for different types of servers and automatically provision the servers, saving time and ensuring that operating system builds are consistent. By using the Patch Management feature, users can establish polices about how patches are installed. SA uniformly enforces those polices.

The following features are currently available as part of SA among others:

- Operating System Provisioning
- Code Deployment & Rollback
- Configuration Tracking
- Script Execution
- Discovery and Agent Deployment
- Device Explorer
- Virtualization Director
- Server Automation Visualizer (SAV)
- Storage Visibility and Automation
- Audit and Remediation
- Compliance View
- Reports
- Software Management
- Patch Management for Windows
- Patch Management for Unix
- Patch Management for Solaris
- OS Provisioning
- Application Configuration Management
- Global Shell

All SA features support cross-platform environments and are designed to automate both new and existing data center environments.

Operating System Provisioning

The OS Provisioning feature gives administrators the ability to provision operating system baselines onto bare metal servers quickly, consistently, and with minimal manual intervention. Bare metal OS provisioning is a key part of the overall process of getting a server into production.

Benefits of the OS Provisioning feature include the following items:

- **Integration with the other features of SA**

  Because the OS Provisioning feature is integrated with the suite of SA automation capabilities, including patch management, software management, and distributed script execution, hand-off between IT groups are seamless. SA ensures that all IT groups are working with a shared understanding of the current state of the environment, which is an essential element of delivering high-quality operations and reliable change management.

- **The ability to easily update server baselines without re-imaging servers**

  Unlike many other OS provisioning solutions, systems provisioned with SA can be easily changed after provisioning to adapt to new requirements. The key to this benefit is the SA use of templates and its installation-based approach to provisioning.

- **Flexible architecture designed to work in many environments**

  SA engineers carefully designed the OS Provisioning feature to handle many different types of servers, networks, security architectures, and operational processes. SA works well in floppy (Windows provisioning), CD (Linux provisioning), or network-boot environments, with scheduled or on-demand workflows, and across a large variety of hardware models. This flexibility ensures that you can provision operating systems to suit your organization's needs.

SA automates the entire process of provisioning a comprehensive server baseline, which typically consists of the following tasks:

- Preparing the hardware for OS installation using an OS installation profile
- Creating OS sequences that define a server build policy, including application policies, patch policies, device groups, and remediation policies
- Installing a base operating system and default OS configuration using an OS sequence
- Applying the latest set of OS patches, the exact list depends on the applications running on the server
- Installing system agents and utilities such as SSH, PC Anywhere, backup agents, monitoring agents, or anti-virus software
- Installing widely-shared system software such as Java Virtual Machines
- Executing pre-installation or post-installation scripts that configure the system with values such as a root password

Code Deployment & Rollback

SA automates code and content deployment to reduce the risk and time requirements associated with pushing new code to production. The Code Deployment & Rollback (CDR) feature provides an automated system for deploying code (such as, ASP, JSP, JAR, Java, C++, and Perl files) and content (such as, HTML, JPEG, GIF, and PDF files).

Specifically, CDR enables you to perform the following actions:

- Push code from staging or development environments to production environments.
- Synchronize code and content across multiple servers and locations.
- Automatically rollback to the previous version of code or content.
- Sequence multiple, complex deployment steps into repeatable workflows.
- Manage changes across heterogeneous operating systems.

Configuration Tracking

The Configuration Tracking feature tracks, backs up, and recovers critical software and system configuration information across Unix and Windows servers.

System administrators set up policies that describe the configuration files and databases to track, and the actions to take when a change in configuration is detected. Policies can be assigned to software, individual servers, groups of servers, and customers, and applied either locally or globally across data centers.

When SA notices a server configuration change, it can log the change, notify administrators about the change with email, or back up the configuration, depending on the policy set by the administrator.

When a bad configuration change forces administrators to rollback to a previous version, they can use SA to restore the configuration file to the saved version of the configuration. By notifying users about configuration changes — and maintaining a version history of those changes — organizations can quickly diagnose problems related to configuration errors and rollback to a known good state. In addition, this capability helps teams plug security holes inadvertently created by bad server configurations.

Typically, system administrators define configuration-tracking policies on a per-application basis. So for example, a policy for BEA WebLogic might specify, “Monitor the `weblogic.conf` file, notify `app-server-admins@company.com` of any changes, and maintain a version history of any changes that occur for 30 days.” After a policy is defined in this fashion, administrators can apply the policy to all the WebLogic servers running in their environment or to specific servers.

Script Execution

The Script Execution feature enables you to share and run ad-hoc or saved scripts across an entire farm of SA-managed servers.

By executing scripts with SA instead of manually, administrators benefit by using the following features:

- Parallel script execution across many Unix and/or Windows servers, saving time and ensuring consistency.
- Role-based access control, ensuring only authorized administrators can execute scripts on hosts to which they have access.
- The ability to control access to scripts by storing them in private or in public libraries.
- The ability to see and download script output one server at a time or in a consolidated report, which captures output from all servers in a single place.

- The ability for scripts to be mass-customized. Administrators can access information in SA about the environment and the state of servers. This is critical to ensuring that the right scripts are executed on the right servers.
- A comprehensive audit trail that reports who, what, when, and where a particular script was executed.

Because the Script Execution feature is an integrated part of SA, administrators enjoy unique benefits when compared to standalone script execution tools:

- Using known system state and configuration information to customize script execution, users can tailor each script by referencing and accessing the rich store of information in SA, such as the customer or business that owns the server, whether the server is a staging or production server, which facility the server is located in, and custom name-value pairs.
- By sharing scripts without compromised security, users can share scripts with each other without compromising security because SA maintains strict controls on who can execute scripts on which servers and generates a comprehensive audit trail of script execution.

Discovery and Agent Deployment

The SA Discovery and Agent Deployment (ODAD) feature allows you to deploy Server Agents to a large number of servers in your facility and place them under SA management.

Using the ODAD features, you can perform the following tasks:

- Scan your network for servers.
- Select servers for SA Agent installation.
- Select a communication tool and provide user/password combinations.
- Choose agent installation options and deploy agents.

Device Explorer

The Device Explorer lets you view information about servers in your managed environment.

From the Server Explorer, you can perform the following tasks:

- Create a server snapshot, perform a server audit, audit application configurations, create a package, and open a remote terminal session on a remote server.
- Browse a server's file system, registry, hardware inventory, software and patch lists, and services.
- Browse SA information such as properties, configurable applications, and even server history.

From the Groups Browser, you can perform the following tasks:

- Audit system information, take a server snapshot, and configure applications.
- View and access group members (servers and other groups).
- View group summary and history information.

Virtualization Director

The Virtualization Director feature enables you to provision and manage virtual servers for Solaris 10 local zones and VMware ESX 3 virtual machines (VMs). Using the SA Client, you can perform the following tasks:

- View both hypervisor and virtual servers and their relationships in the SA Client, so you can find out the hypervisors that are hosting your virtual machines and local zones.
- View virtual servers and their relationship in the HP Server Automation Visualizer (SAV).
- Provision VMware ESX and Solaris 10 hypervisors on bare metal servers.
- Provision VMware virtual machines (VM) using an OS sequence.
- Create, start, stop, modify, and remove Solaris local zones.
- Deploy agents on unmanaged virtual servers using the Agent Discovery and Deployment for VMware ESX VMs.
- Search for virtual servers in your data center using the Search tool.
- Create dynamic Device Groups based upon virtual server characteristics (zones or VMs).

Server Automation Visualizer (SAV)

The Server Automation Visualizer (SAV) feature is designed to help you optimally understand and manage the operational architecture and behavior of distributed business applications in your IT environment. Since these applications are complex collections of services that typically run across many servers, as well as network and storage devices, it can become increasingly difficult to understand (or remember) what is connected to what, where performance problems originate, how to troubleshoot and resolve problems, and what result would occur if you make a change in your environment.

SAV helps you see (visualize) this type of information through physical and logical drawings.

Storage Visibility and Automation

The Storage Visibility and Automation feature offers storage management capabilities by enabling end-to-end visibility and management of the entire storage supply chain. This feature helps server administrators day-to-day tasks by providing tools that increase cost savings through application storage, dependency and visibility, storage audits, storage capacity and utilization trending, and scripting and automation. See the Storage Visibility and Automation User's Guide for more information.

Audit and Remediation

The Audit and Remediation feature allows you to define server configuration policies and make sure that servers in your facilities meet those policy standards. When servers are found to be ‘out of compliance’ (not configured the way you want them to be), you can remediate the differing server configurations.

With Audit and Remediation, you can audit a server configuration values based upon a live server (or server snapshot), or based upon your own custom values, perform server comparisons against a baseline, and create custom audit policies that define company or industry server configuration compliance standards, and which can be used inside of audits, snapshot specifications, and audit policies.

Using Audit and Remediation, you can perform the following tasks:

- Compare servers or snapshots to reference servers or snapshots
- Create audits for repeated use
- Create audit policies that define compliance and security standards for your organization
- Associate audits with individual servers or dynamic server groups
- Remediate problems at multiple levels, including files, directories, patches, registry keys, and packages

Compliance View

The Compliance Dashboard allows you to view at a glance the overall compliance levels for all the devices in your facility and helps you to remediate compliance problems. The Compliance Dashboard displays compliance tests for software policies, application configurations, audits, patches, and duplex status. Each of these compliance tests is based upon an HP Server Automation "policy" (user or system defined) which define a unique set of server or device configuration settings or values that help ensure your IT environment is configured the way you want it to be.

Reports

The Reports feature provides comprehensive, real-time information about managed servers, network devices, software, patches, customers, facilities, operating systems, compliance policies, and users and security in your environment. These reports are presented in graphical and tabular format, and are actionable—where you can perform appropriate actions on objects, such as a policy or an audit, within the report. These reports are also exportable to your local file system (in .html and .xls formats) to facilitate use within your organization.

Software Management

The Software Management feature in HP Server Automation provides a powerful mechanism to model software by using software policies and to automate the process of deploying software and configuring applications on a server in a single step. In addition, the Software Management feature provides a structure to organize your software resources in folders and define security permissions around them. This feature allows you to verify the compliance status of a server and remediate non-compliant servers.

The Software Management feature in SA Client provides the following functions:

- Create an organizational structure for software
- Define security boundaries for folders
- Define a model-based approach to manage the IT environment in your organization
- Enable sharing of software resources among user groups

- Deploy and configure applications simultaneously
- Deploy multiple application instances on one server
- Establish a software deployment process
- Verify compliance status of servers to software policies
- Generate reports
- Comprehensively search for software resources and servers

Patch Management for Windows

The Patch Management for Windows feature enables you to identify, install, and remove Microsoft® Windows patches and maintain a high level of security across managed servers in your organization. With SA Client user interface, you can identify and install patches that support security vulnerabilities for the Windows 2000, Windows 2003, and Windows NT4.0 operating systems. These patches include Service Packs, Update Rollups, and hotfixes.

Patch Management for Unix

The Patch Management for Unix feature enables you to identify, install, and remove Unix patches to maintain a high level of security across managed servers in your organization. With the SA Client, you can identify and install patches that support security vulnerabilities for the AIX, HP-UX, Linux, and Solaris operating systems.

Patch Management for Solaris

HP Server Automation patch management for Solaris allows you to automate the process of installing and uninstalling Solaris patches and patch clusters on Sun Solaris using patch policies. In addition, SA analyzes the dependency, supersedence, and applicability relationships between patches in the policy and displays an updated and ordered list of patches that should be installed on the server. This feature allows you to verify the compliance status of a server and remediate non-compliant servers and automatically download the Solaris patches into SA and organize them into patch policies.

OS Provisioning

OS Provisioning in the SA Client allows you to install an operating system, applications, and packages and packages on unprovisioned servers by creating OS Installation Profiles and OS Sequences. From the Devices list, you can view all unprovisioned servers in your facility and provision those servers by running an OS Sequence.

OS Sequences allow you to set up a complete server build (policy) that represents the ideal manner in which a particular OS should be installed, which includes the proper OS Installation Profile that should be used, as well as Application and Patch policies, the servers to install the OS on, and how these policies should be remediate either before or after the OS is installed.

Application Configuration Management

Application Configuration Management (ACM) allows you to create templates so you can modify and manage application configurations associated with server applications. ACM enables you to manage, update, and modify those configurations from a central location, ensuring that applications in your facility are accurately and consistently configured.

Using ACM, you can perform the following tasks:

- Manage configurations based on files and objects, such as Windows registry, IIS metabase, WebSphere, COM+, and more.
- Preview configuration changes before applying them.
- Edit and push configuration changes to individual servers or server groups.
- Use information in the SA data model to set configuration values.
- Manage configurations of any application by building configuration templates.
- Audit the Application Configurations on a server to determine if any of the configuration files on the server are out of sync with the values stored in your templates.

Global Shell

The Global Shell feature enables you to manage servers by using a command-line interface. You can remotely perform the following tasks:

- Complete routine maintenance tasks on managed servers.
- Troubleshoot, identify, and remediate problems on managed servers.

Global Shell consists of a file system and a command-line interface to that file system for managing servers in SA. The file system is known as the SA Global File System (OGFS). All object types in the OGFS (such as servers, customers, and facilities) are represented as directory structures in this file system.

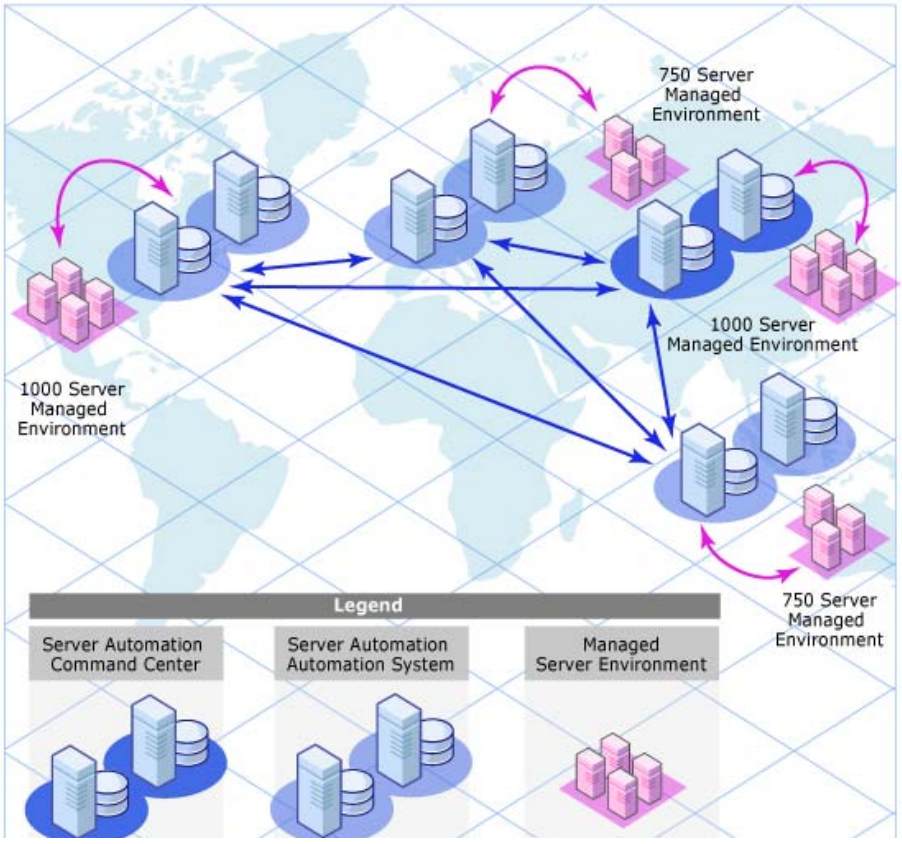
The Global Shell feature also manages user permissions for accessing the file system, Windows Registry, and Windows Services objects on managed servers.

3 Advanced SA Architecture and Options

This section is intended for those users who want a more in-depth understanding of SA architecture because they intend to customize the layout of their SA cores, create a Multimaster Mesh, require a remote database installation, and so on. You will learn about the SA Core and its Core Components and the relationship between the core, Server Agents, and Satellites.

Figure 4 is an example of an advanced Multimaster Mesh installation in which core components are distributed on different host and there are multiple SA Cores with their managed servers in different locations, each communicating with the other.

Figure 4 An Advanced Multimaster Installation



The SA Core

An *SA Core* is a set of *Core Components* that work together to allow you to discover servers on your network, add those servers to a Managed Server Pool, and then provision, monitor, configure, audit, and maintain those servers from a centralized SAS Web Client or SA Client interface. These clients provide a GUI interface to the information and management capabilities of SA.

The servers that the Core Components are installed on are called *Core Servers*. Core Components, even if distributed to multiple hosts are still considered part of a single SA Core.

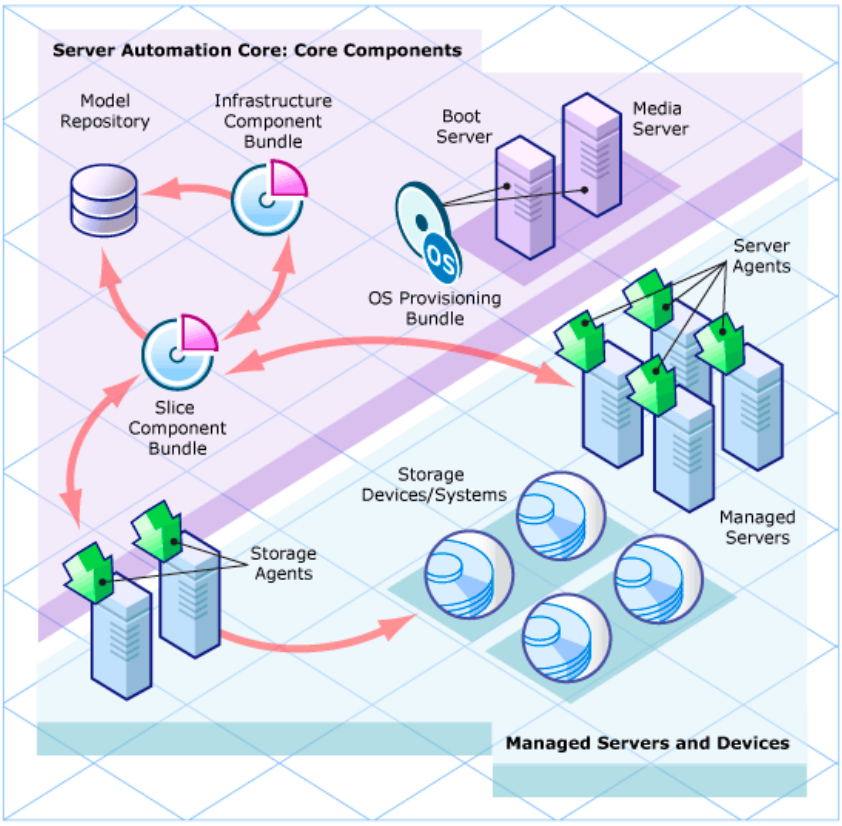
Core Components can all be installed on a single host or distributed across several hosts, however, the typical SA installation uses *Core Component bundling* which installs certain components together on the same server for performance and maintainability purposes. See SA Core Component Bundling on page 26 for more information about component bundling.

In order to communicate and perform certain server management activities, SA installs *Server Agents* on each Managed Server and communicates with the Managed Servers through Gateways that are part of the SA Core Components. Server Agents also perform certain actions on Managed Servers as directed by user input from the SA Client or SAS Web Client.

A Simple Single Core Installation

Figure 5 shows a simplified representation of a single core with all Managed Servers in the same facility, typically the First Core of a Multimaster Mesh. Most installations consist of multiple cores in different facilities. See SA Topologies on page 34.

Figure 5 An *SA* Core and Agents



A *Core Server* hosts the SA Core Components that allow SA to discover and store information about the location and configuration of all the servers on your network as well as components that perform monitoring, auditing, provisioning and maintenance tasks.

SA Server Agents

An SA Server Agent is intelligent software that is installed on a server that you want to manage using SA. After an agent is installed on an unmanaged server, it registers with the SA Core which can then add that server to its pool of Managed Servers. The Server Agent also receives commands from the Core and initiates the appropriate action on its local server, such as software installation and removal, software and hardware configuration, server status reporting, auditing, and so on.

You can install agents on servers in the following ways:

- You can use the SA Deployment and Discover (ODAD) utility to discover the servers on your network that do not have SA Server Agents installed and then deploy the agents to those servers.
- You can use the SA OS Provisioning feature to provision an operating system to a bare-bones server — an SA Server Agent will also be installed.
- You can copy the SA Server Agent binary to the server and install it manually.

During agent registration, SA assigns each server a unique ID (the Machine ID (MID)) and stores this ID in the Model Repository. Servers can also be uniquely identified by their MAC Address (the network interface card's unique hexadecimal hardware identifier, which is used as the device's physical address on the network).

The Core Components

The Core Components are the heart of the SA Core making it possible to communicate with, monitor, and manage servers. Users and developers interact with the core through the SA Client or SAS Web Client, the command line, the API, and so on. Users can retrieve vital information about their network servers, provision servers, apply patches, take servers on and off line, configure and audit servers, and more. This interaction is controlled by the Core Components.

For example, a user can use OS Provisioning of the SA Client to remotely install an operating system on a server and bring it under SA management.

The following section describes the SA Core Components and interfaces. For detailed information about how the SA Components work together to manage your servers, see the *SA Administration Guide*.

SA Core Component Bundling

Certain SA Core Components are *bundled* together and must be installed as a *unit* during a Typical Installation. It is possible, if necessary, to break certain components (such as the Repository Store, OS Provisioning Media Server, OS Provisioning Boot Server, among others) out of a bundle to install them on a different host by performing a Custom installation.

For more information about Typical vs. Custom installations, see the *SA Simple/Advanced Installation Guide*.

In addition, the Slice Component bundle can have multiple installations on multiple hosts.

SA Core Component bundling provides the following benefits:

- Added simplicity and robustness for multi-server deployments
- Scaling capability: you can install additional Slice Component bundles for horizontal scaling
- Improved High Availability
- Load balancing between slices when multiple instances installed

New in SA 7.50:

- The SA Command Engine is installed as part of the Slice Component bundle, therefore you can have multiple Command Engine per core thus increasing SA 7.50's ability to manage large numbers of servers simultaneously.

New in SA 7.80:

- The *Software Repository* is now installed as part of the *Slice Component bundle*, therefore you can have multiple Software Repositories per Core. Also new is the *Software Repository Store* which is part of the *Infrastructure Component bundle* and handles NFS exports to Slice Component bundle hosts.

New in SA 9.0

- No Core Component architectural changes
- Software Repository mirroring enabled by default in Multimaster Meshes

Table 1 shows how SA Core Components are bundled.

Table 1 Component Distribution

Model Repository	Infrastructure Components	OS Provisioning Components	Slice Components#1	Slice Components#x
One per core	One per core	Typically one per core	One per core	Multiple per core
Model Repository	Management Gateway Primary Data Access Engine Model Repository Multimaster Component Software Repository Store (can be located on another host)	Media Server Boot Server	Core Gateway/ Agent Gateway Command Center Global File System Web Services Data Access Engine Secondary Data Access Engine Build Manager Command Engine Software Repository HP Live Network (HPLN) DCML Exchange Tool (DET)	Core Gateway/ Agent Gateway Command Center Global File System Web Services Data Access Engine Secondary Data Access Engine Build Manager Command Engine Software Repository HP Live Network (HPLN) DCML Exchange Tool (DET)

The *Boot Agent* is unrelated to Server Agents and operates as part of OS Provisioning.

Model Repository

The Model Repository requires either the SA-supplied Oracle database or an existing Oracle installation that must meet SA database requirements. For more information about these requirements see the *SA Advanced Installation Guide* or the document, *Oracle Setup for the Model Repository.*

The Model Repository is a standalone component and is not bundled with other Core Components. All SA components work from or update a data model maintained for all SA Managed Servers. The Model Repository stores the following information:

- An inventory of all servers under SA management.
- An inventory of the hardware associated with these servers, including memory, CPUs, storage capacity, and so on.
- Information about managed server configuration.
- An inventory of the operating systems, system software, and applications installed on managed servers.

- An inventory of OS Provisioning operating system installation media (the media itself is stored in the OS Provisioning Media Server).
- An inventory of software available for installation and the software policies that control how the software is configured and installed. The software installation media itself is stored in the Software Repository.
- Authentication and security information.

The Core Component Bundles

Infrastructure Component Bundle

- **Primary Data Access Engine**

  The Data Access Engine provides an XML-RPC interface to the Model Repository that simplifies interaction with various clients, such as the SAS Web Client, system data collection, and monitoring agents on servers. The Data Access Engine installed with the Infrastructure Component bundle is designated the *Primary* Data Access Engine. The Data Access Engine installed with the Slice Component bundle(s) is designated the *Secondary* Data Access Engine.

  Because interactions with the Model Repository go through the Data Access Engine, clients are less impacted by changes to the Model Repository's schema. The Data Access Engine allows features to be added to SA without requiring system-wide changes.

- **Management Gateway**

  Manages communication between SA Cores and between SA Cores and Satellites.

- **Model Repository Multimaster Component**

  The Model Repository Multimaster Component is installed with the Infrastructure Component bundle. A Multimaster Mesh, by definition, has multiple core installations and the Model Repository Multimaster Component synchronizes the data in the Model Repositories for all cores in the Mesh, propagating changes made in one repository to the other repositories.

  Each Model Repository Multimaster Component consists of a Sender and a Receiver. The Sender (Outbound Model Repository Multimaster Component) polls the Model Repository and sends unpublished transactions to other Model Repositories. The Receiver (Inbound Model Repository Multimaster Component) accepts the transactions from other Model Repositories and applies them to the local Model Repository.

As of SA 7.80, TIBCO Rendezvous was replaced by the SA Bus. The SA Bus is a set of libraries that provide certified messaging services.

- **Software Repository Store**

  The Software Repository Store component can be installed on any server hosting an Infrastructure Component bundle. As of SA 9.0, the Software Repository is part of the Slice Component bundle and the Software Repository Store component has been introduced to handle NFS exports to Slice Component bundle hosts.

  If you choose not to install the Software Repository Store, you must manually configure a NAS (filer) to allow Slice Component bundle servers access to the file system.

Slice Component Bundle

- **Command Engine**

  Part of the Slice Component bundle. The Command Engine is a system for running distributed programs across many servers (typically through SA Server Agents). Command Engine scripts are written in Python and run on the Command Engine server. Command Engine scripts can issue commands to Server Agents. These calls are delivered in a secure manner and are auditable by using data stored in the Model Repository.

  Because you can have multiple Slice Component bundles, and therefore multiple Command Engines, horizontal scaling is greatly enhanced. Multiple Command Engine instances can share the load of command delivery and script execution by taking advantage of the load balancing mechanism provided by multiple Slice Component bundles. Failover and high availability are also improved. For example, when a Command Engine instance tries to delegate a command to another node in the cluster and that node is down, it fails over to the next node.

  SA features (such as Code Deployment & Rollback) can use Command Engine scripts to implement part of their functionality.

- **Software Repository**

  Part of the Slice Component bundle. This component is a repository in which the binaries/packages/source for software/application provisioning and remediation is uploaded and stored. A related component is the Software Repository Store which is installed with the Infrastructure Component bundle and handles NFS exports to Slice Component bundle hosts.

  SA supports mirroring of the Software Repository. You can control which Software Repositories in the mesh are designated as mirrors and control the frequency of mirroring jobs by modifying configuration parameters in the SAS Web Client. Mirroring does not affect Satellite Software Repository caches.

  Software Repository mirroring can require large amounts of available disk space. During Standard and Advanced installation, you are given the opportunity to turn off mirroring which is on by default.

  For more information about configuring Software Repository mirroring, see the *SA Administration Guide*.

  For information about how to upload software packages to the Software Repository, see the *SA Policy Setters Guide*.

- **Core Gateway/Agent Gateway**

  The Core Gateway communicates directly with Agent Gateways passing requests and responses to and from Core Components.

- **Command Center**

  The Command Center (OCC) is the Core Component that underlies the SAS Web Client. The OCC includes an HTTPS proxy server and an application server. You access the OCC only through the SAS Web Client.

- **DCML Exchange Tool**:

  The DCML Exchange Tool is installed with each Slice Component bundle and facilitates the import and export of SA content. See the *SA Content Utilities Guide*.

- **Global File System**

  The Global File System (OGFS) is installed with each Slice Component Bundle and provides the central execution environment for SA.

  The OGFS runs on one or more physical servers; customers can scale SA execution capacity by simply adding additional Slice Component bundles in a core.

  The OGFS runs SA built-in components — as well as customer-written programs — within a virtual file system that presents the SA data model, SA actions, and managed servers as virtual files and directories.

  This unique feature of SA allows users of the Global Shell and Automation Platform Extensions (APX) to query SA data and manage servers from any scripting or programming language. Since the OGFS filters all data, actions, and managed server access through the SA security model, programs running in the OGFS are secure by default.

- **Web Services Data Access Engine**

  The Web Services Data Access Engine provides a public-object abstraction layer to the Model Repository and provides increased performance to other SA Core Components. This object abstraction can be accessed through a Simple Object Access Protocol (SOAP) API, through third-party integration components, or by a binary protocol of SA components such as the SAS Web Client.

- **Secondary Data Access Engine**

  The Data Access Engine provides an XML-RPC interface to the Model Repository that simplifies interaction with various clients, such as the SAS Web Client, system data collection, and monitoring agents on servers. The Data Access Engine installed with the Infrastructure Component bundle is designated the *Primary* Data Access Engine. The Data Access Engine installed with the Slice Component bundle(s) is designated the *Secondary* Data Access Engine.

  Because interactions with the Model Repository go through the Data Access Engine, clients are less impacted by changes to the Model Repository's schema. The Data Access Engine allows features to be added to SA without requiring system-wide changes.

- **Build Manager**

  Although the Build Manager is part of the OS Provisioning feature it is installed as part of the Slice Component bundle. The Build Manager facilitates communications between OS Build Agents and the Command Engine. It accepts OS Provisioning commands from the Command Engine. It provides a runtime environment for the platform-specific build scripts to perform the OS Provisioning procedures.

- **HP Live Network (HPLN)**

  HP Live Network delivers content updates for Server Automation (SA), Network Automation (NA), Client Automation (CA), Operations Orchestration (OO) and Service Automation Reporter (SAR). The HP Live Network (HPLN) provides customers with security and compliance policies to help maximize your return on investment in SA, NA, and CA, and to leverage the extensible automation platforms to deliver new automation capabilities on an ongoing basis.

  HPLN is installed as part of the Slice Component bundle during SA Core installation.

OS Provisioning Components Bundle

- **Boot Server**

  The Boot Server is part of the OS Provisioning feature. It supports network booting of Sun and x86 systems with inetboot and PXE, respectively. The processes used to provide this support include the Internet Software Consortium DHCP server, Sun Solaris TFTP, and NFS.

- **Media Server**

  The Media Server is part of the OS Provisioning feature. It is responsible for providing network access to the vendor-supplied media used during OS Provisioning. The processes used to provide this support include the Samba SMB server and Sun Solaris/Linux NFS. You copy and upload your valid operating system installation media to the Media Server.

OS Build Agent: The OS Build Agent is part of the OS Provisioning feature. It runs during the pre-provisioning (network boot) process and is responsible for registering a bare metal server with the SA Core through the Build Manager and guiding the OS installation process.

Satellite Installations

- **Software Repository Cache**

  A Software Repository Cache contains local copies of the contents of a Core's Software Repository (or of another Satellite). Having a local copy of the Software Repository can improve performance and decrease network traffic when you install or update software on a Satellite's Managed Servers.

- **Satellite Agent Gateway**

  The Satellite Agent Gateway handles communications between the Satellite and the Core through the Core's Management Gateway.

SA Interfaces

SAS Web Client

The SAS Web Client is an HTML browser-based user interface to SA through which users can:

- Manage servers
- Deploy code and content to servers (deprecated)

SA Client

A Java™ Web-Start Windows application that extends the SAS Web Client features and provides the following features:

- Configuring Software Policies
- Provisioning software/applications/packages onto Managed Servers
- Provisioning operating systems onto bare metal servers

- Running distributed scripts on servers
- Unmanaged server Discovery and Agent Deployment (ODAD)
- Detailed hardware information using Device Explorer
- Virtual server management
- Managing the operational architecture and behavior of your distributed business applications using Service Automation Visualizer (SAV)
- Audit and Remediation
- Compliance Dashboard
- Reports
- Software Management
- Patch Management for Windows, Unix, Solaris, HP-UX
- Application Configuration Management
- Global File System and Global Shell access for command line server management
- Network Automation and Job Approval integration

Automation Platform Extensions

APXs provide a framework that allows anyone familiar with script-based programming tools such as shell scripts, Python, Perl, and PHP, to extend the functionality of SA and create applications that are tightly integrated into SA.

SA Command Line Interface (OCLI)

A command line interface used to upload packages into the Software Repository, and to perform batch commands, run scripts, and many other SA operations.

DCML Exchange Tool (DET)

A utility that enables users to export almost all server management content from any SA Core and import it into any other SA Core.

ISM Development Kit

A development kit that consists of command-line tools and libraries for creating, building, and uploading ISMs. An ISM is a set of files and directories that include application bits, installation scripts, and control scripts.

SA APIs

A set of APIs and a command-line interface (CLI) that facilitate the integration and extension of SA. This platform allows other IT systems — such as existing monitoring, trouble ticketing, billing, and virtualization technology — to exchange information with SA. This broadens the scope of how IT can use SA to achieve operational goals.

For more information about all the interfaces, see the *SA Administration Guide*.

SA Gateways

SA Gateways manage communication between Managed Servers and a SA Core, between multiple cores, and between Satellite installations and a SA Core. Multimaster installations are discussed in Multimaster Mesh (Multiple Cores) on page 35 and Satellite installations are discussed in Multimaster Mesh (Multiple Cores and Satellites) on page 36.

There are several types of gateways:

- **Management Gateway**

  This gateway manages communication between SA Cores and between SA Cores and Satellites.

- **Core Gateway/Agent Gateway**

  These gateways work together to facilitate communication between the SA Core and Server Agents.

- **Satellite Gateway**

  This gateway communicates with the SA Core through the Management Gateway or the Core Gateway depending on your configuration.

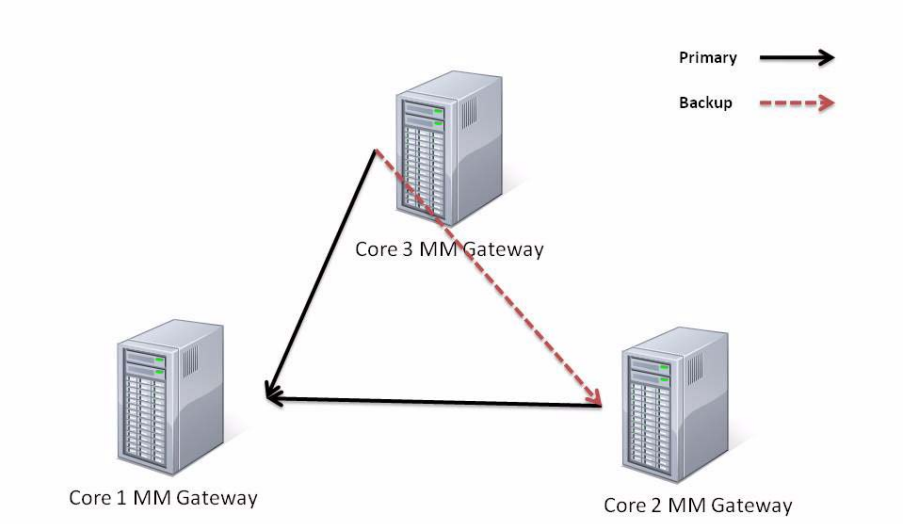
Multimaster Master Gateway Backup Routes

With this SA release, by default, installation of a third or subsequent core in a Multimaster Mesh automatically creates a backup route to the Second Core providing a primary route to the First Core and a backup route to the Second Core. SA creates the Gateway backup routes automatically during installation, you are not required to provide any configuration information, however, if SA cannot create the backup routes, you will see a message to that effect and may need to contact HP Technical Support to manually configure Gateway backup routes.

Gateway backup routes are created only during fresh installations of SA 9.0, not during upgrades. If you are upgrading to SA 9.0 from an earlier version, the upgrade will not create gateway backup routes. You must create backup routes manually. Contact your HP Technical Support representative for more information.

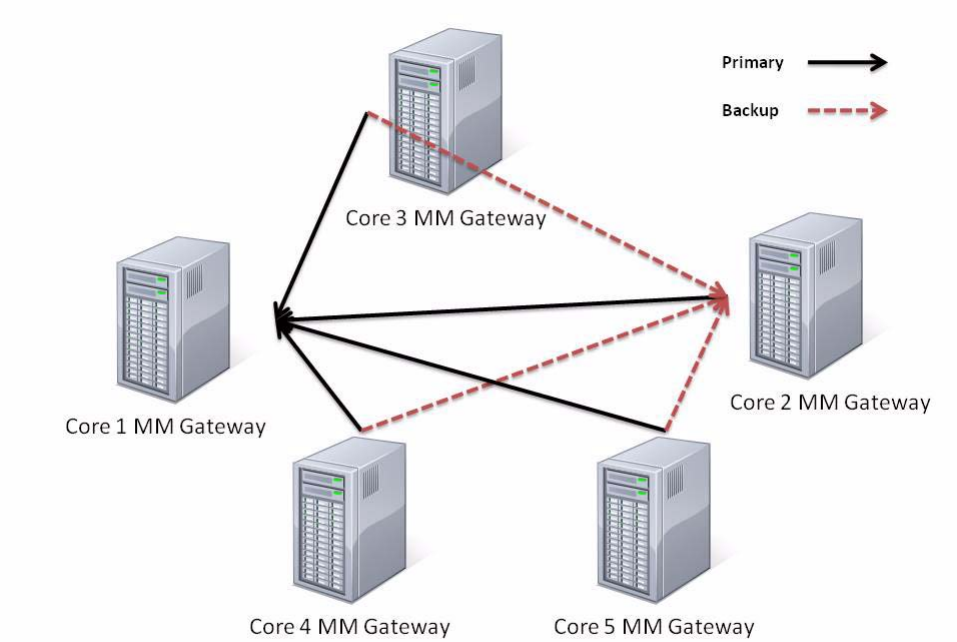
For example, for a three core or more mesh, all Multimaster traffic is routed by default through the First Core's Master Gateway. However, the Second Core's Master Gateway is now designated as the by default as the backup Master Gateway should the First Core's Master Gateway fail. All additional subsequent core's Master Gateways added to the mesh will be designated as a backups in the order of installation. On installation, the third and subsequent cores will by default have two tunnels. The first tunnel communicates with the First Core's Master Gateway, the second tunnel with the second core in the mesh. SeeFigure 6.

Figure 6 Mesh with Three Cores and a Single Backup Route



A mesh with multiple Master Gateways will also have redundant backup routes. See Figure 7.

Figure 7 Five Core Mesh with Multiple Backup Routes



Upon failure of a Master Gateway, the backup route will automatically be used for Multimaster Mesh traffic by default. When the failed Master Gateway is brought back on line, mesh traffic will automatically be routed through that gateway again.

SA Topologies

You must decide what SA topology fits your facility's needs. This section provides some background on the SA topologies to help you make that decision

Single Core

The simplest topology is a Single Core (formerly a Standalone Core) that manages servers in a single facility.

A Single Core is best for a small network of servers contained in a single facility. Although a Single Core does not communicate with other SA Cores, it has all the components required to do so and can be easily converted into a core that is part of a Multimaster Mesh.

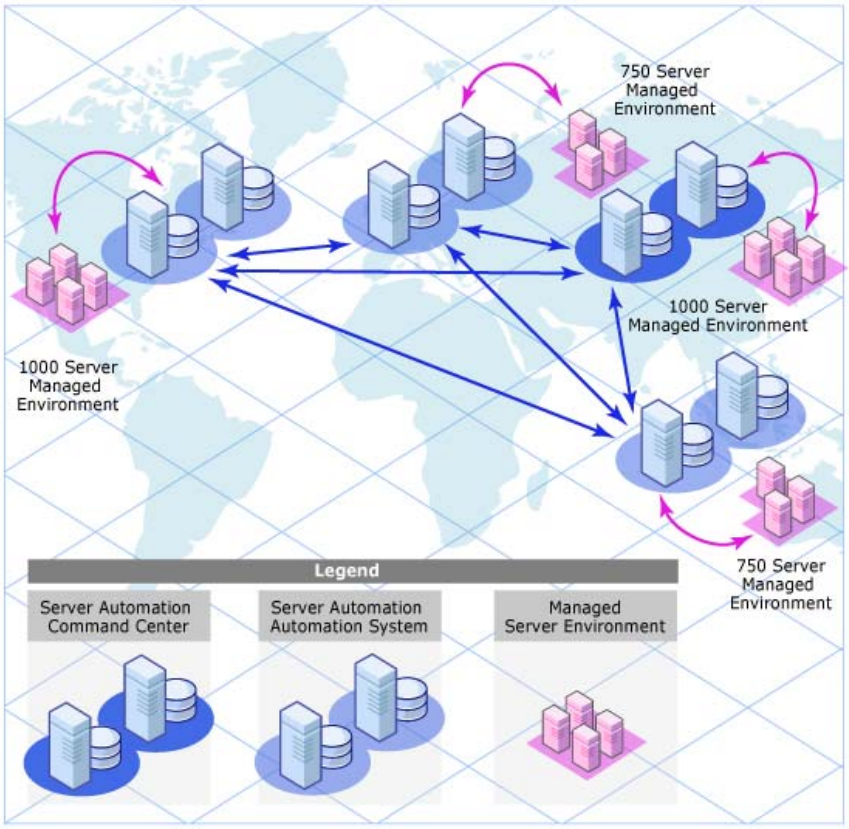
After the core is installed, you can use the Deployment and Discover (ODAD) utility to discover the servers on your network that do not have Server Agents installed and then deploy agents to those servers. After the Server Agents are deployed, they will automatically contact the Core through the Agent Gateway and register the server they are installed on with SA.

You can then use the SA Client to manage your servers.

Multimaster Mesh (Multiple Cores)

To manage servers in more than one facility, you should install a Multimaster Mesh of SA Cores or a combination of SA Cores and Satellites.

Figure 8 Multimaster Topology



A *Multimaster Mesh* is a set of two or more SA Cores that communicate through Management Gateways and can perform synchronization of the data about their Managed Servers contained in their respective Model Repositories over the network. Changes to the data in any Model Repository in a Multimaster Mesh are broadcast to all other Model repositories in the Mesh.

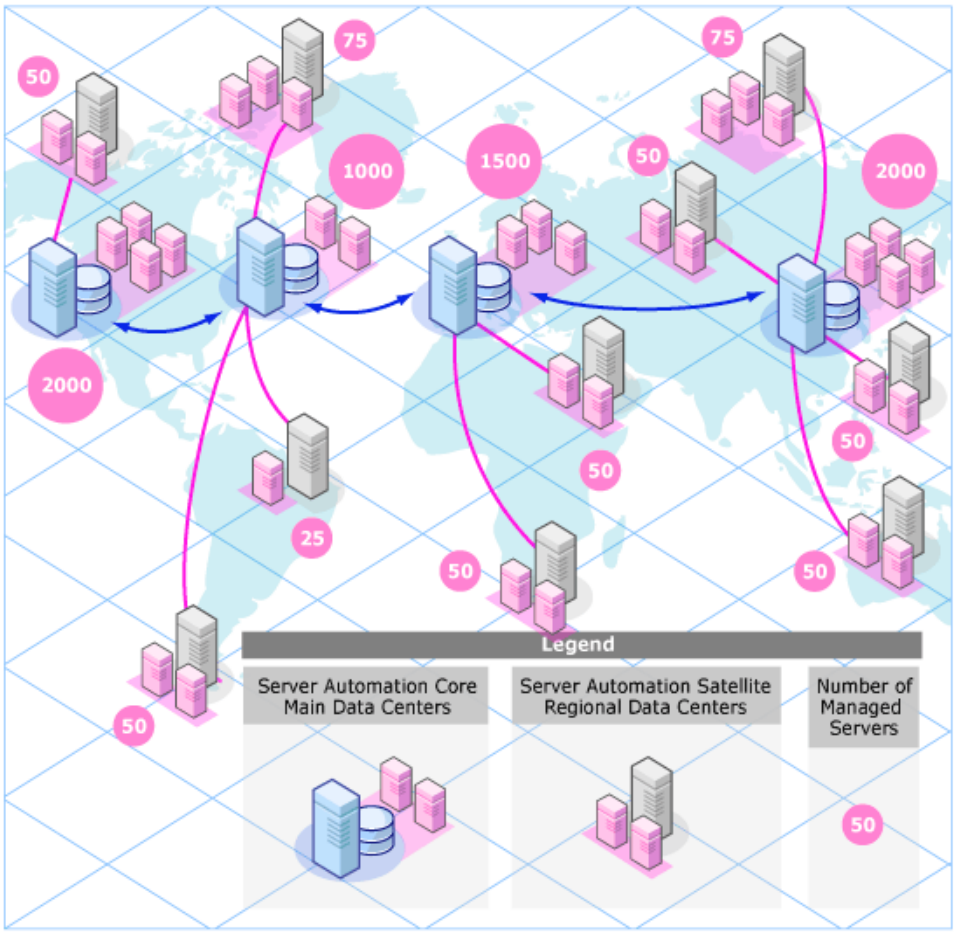
The SA Core Component that propagates and synchronizes changes from each model repository database to all other model repository databases is called the Model Repository Multimaster Component. This replication capability allows you to store and maintain a blueprint of software and environment characteristics for each facility making it easy to rebuild your infrastructure in the event of a disaster. It also provides the ability to easily provision additional capacity, distribute updates, and share software builds, templates and dependencies across multiple facilities — all from a single user interface.

Multimaster Mesh (Multiple Cores and Satellites)

A Multimaster Mesh can also include Satellite installations as shown in Figure 9.

Los Angeles, New York, London, and Tokyo have SA Core installations and each facility links to one or more Satellite installations in smaller facilities some in a star formation, others in cascading Satellite formation. See SA Satellites on page 39.

Figure 9 Server Management in Multiple Facilities with Satellites



Servers can be managed from any facility with an installed SA Core using the SAS Web Client or the SA Client. Using the example in Figure 9, a user can log on to the SA Client at the New York facility and manage servers that belong to the Los Angeles facility as long as he has the appropriate access rights and privileges.

Benefits of Multimaster Mesh

An Multimaster Mesh offers the following benefits among others:

- **Centralized Administration** — the Managed Servers in a Multimaster Mesh can be centrally administered from any facility with a Core installation. Administration is not locked into a single location or even restricted geographically.
- **Redundancy** — Synchronized (replicated) data management between facilities provides redundancy. For example, if the SA Core in one facility is damaged, another core in the Multimaster Mesh will contain a synchronized copy of the managed server data that can be used to restore the damaged core's Model Repository to a last known good state. In addition, while a damaged core is unavailable, other cores in the mesh can continue functioning without interruption.

  Replication also provides the ability to close down or add a facility while other facilities in the mesh continue operations without interruption.
- **Performance Scalability** — In a Multimaster Mesh, only multimaster database synchronizations are transmitted over the network reducing network bandwidth load.
- **Geographic Independence** — Cores can continue to manage servers during network interruptions regardless of location.

Facilities and Realms

SA Gateways use two constructs that facilitate routing network traffic and eliminate the possibility of IP address conflicts:

Facilities

A *Facility* is a construct that typically represents a collection of servers that a single SA core manages through the data about the managed environment stored in its Model Repository. A facility typically represents a specific geographical location, such as Sunnyvale, San Francisco, or New York, or, commonly, a specific data center.

A Facility is a permissions boundary within SA, that is, a user's permissions in one Facility do not carry over to another. Every Managed Server is assigned to a single facility. When a device initially registers with the SA Core, it is assigned to the facility associated with the gateway through which it is registering.

For example, Admin A works in Sunnyvale and is in charge of maintaining server patches. In a Facility framework, Admin A is bound to the Sunnyvale Facility as a user. When Admin A views servers, only those servers that are also bound to the Sunnyvale Facility are displayed. He will not see servers for any other Facility.

There are two types of facilities

- **Core Facilities**

  There is one Core Facility for every SA Core installation.
- **Satellite Facilities**

  A default Facility created when you install a Satellite.

Realms

Realms are a SA concept that allow SA to manage servers on different networks in the same Facility without fear of IP address conflicts.

A Realm is a logical entity that defines an IP namespace *within which* all Managed Server IP addresses must be unique. However, servers that are assigned to a *different Realms* can have duplicate IP addresses and still be uniquely identified within SA by their Realm membership.

Realms are interconnected by gateways in what can be described as a *gateway mesh* — a single interconnected network of SA Gateways.

When you create and name a new Facility during installation, a *default* Realm is also created with the same name as the Facility. For example, when you create the Facility, *Datacenter*, the installation also creates a Realm named *Datacenter*. Subsequent Realms in that facility could be named *Datacenter001*, *Datacenter002*, and so on. IP address in each realm are uniquely identified by the combination of the Realm name and the IP address, eliminating any problem with duplicate IP addresses in the same Facility.

Multimaster Mesh Topology Examples

Figure 10 shows a Multimaster Mesh with cores installed in two separate facilities, San Francisco and Los Angeles. Each facility's core has a Model Repository that contains data about the Managed Servers in both facilities. That data is constantly synchronized (replicated) between both Facilities' Model Repositories. The cores communicate through their respective Management Gateways.

Communication from the Managed Servers in the Los Angeles facility to the San Francisco core travels through the Los Angeles Agent Gateway to the Core Gateway, then to the Los Angeles Management Gateway which then communicates with the San Francisco core through the San Francisco Management Gateway and Core Gateway.

Figure 10 Multimaster Mesh with Two Cores

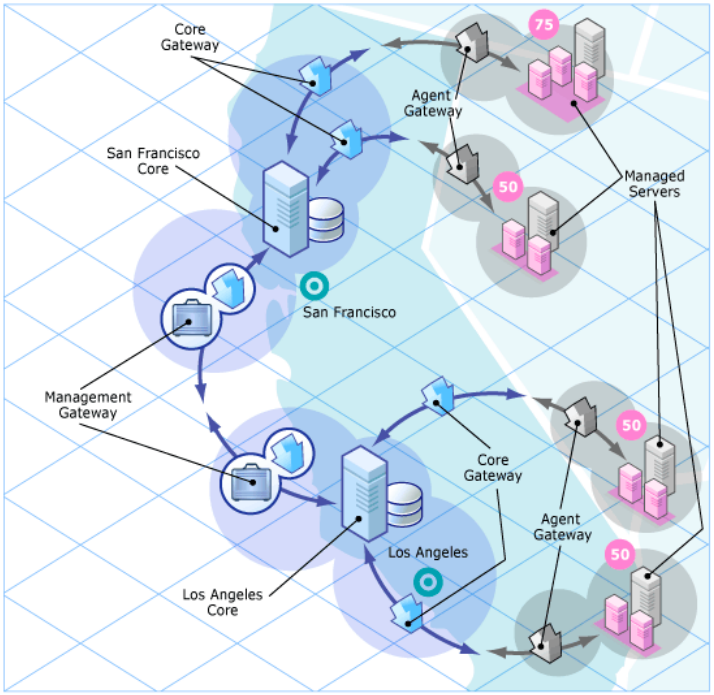
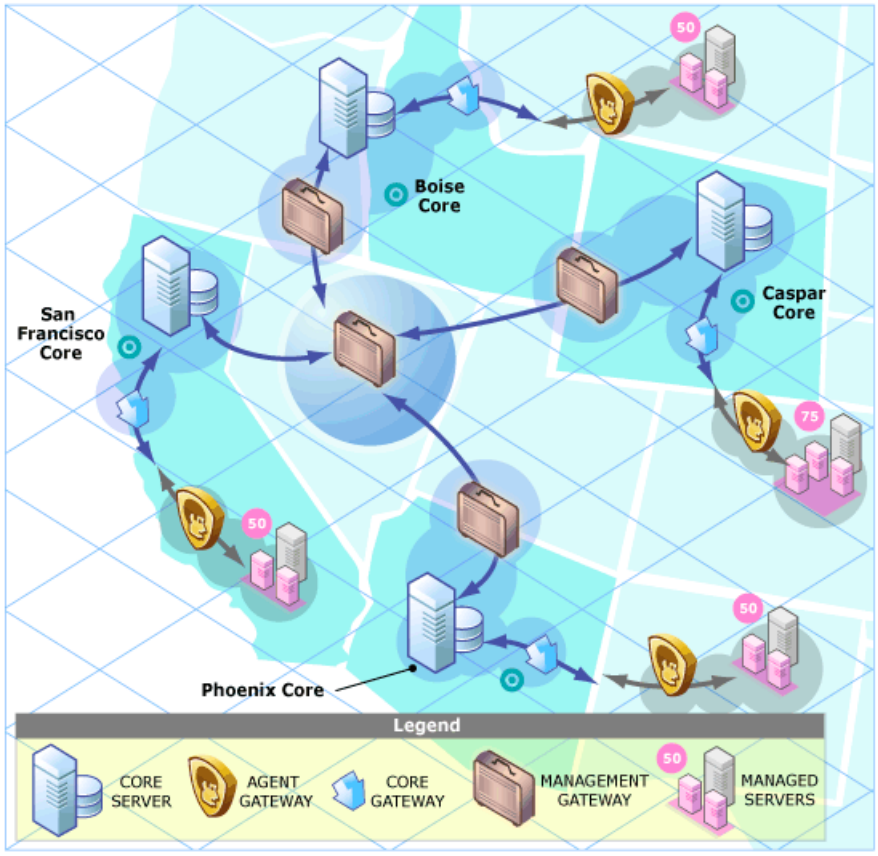


Figure 11 shows a Multimaster Mesh with four cores. This Mesh topology is called a *Star Formation* with the San Francisco core at the center of the Mesh. The SA Installer configures a Multimaster Mesh in a star topology with backup gateway routes by default.

Figure 11 Multimaster Mesh with Four Cores



SA Satellites

A Satellite installation can be a solution for remote sites that do not have a large enough number of potentially Managed Servers to justify a full SA Core installation. A Satellite installation allows you to install only the minimum necessary Core Components on the Satellite host which then accesses the Primary Core's database and other services through an SA Gateway connection.

A Satellite installation can also relieve bandwidth problems for remote sites that may be connected to a primary facility through a limited network connection. You can cap a Satellite's use of network bandwidth to a specified bit rate limit. This allows you to insure that Satellite network traffic will not interfere with your other critical systems network bandwidth requirements on the same pipe.

A Satellite installation typically consists of, at minimum, an Satellite Gateway and a Software Repository Cache and still allows you to fully manage servers at a remote facility. The Software Repository Cache contains local copies of software packages to be installed on Managed Servers in the Satellite while the Satellite Gateway handles communication with the Primary Core.

You can optionally install the OS Provisioning Boot Server and Media Server on the Satellite host to support remote OS Provisioning. Installing other components on the Satellite host is not supported.

Satellite Topology Examples

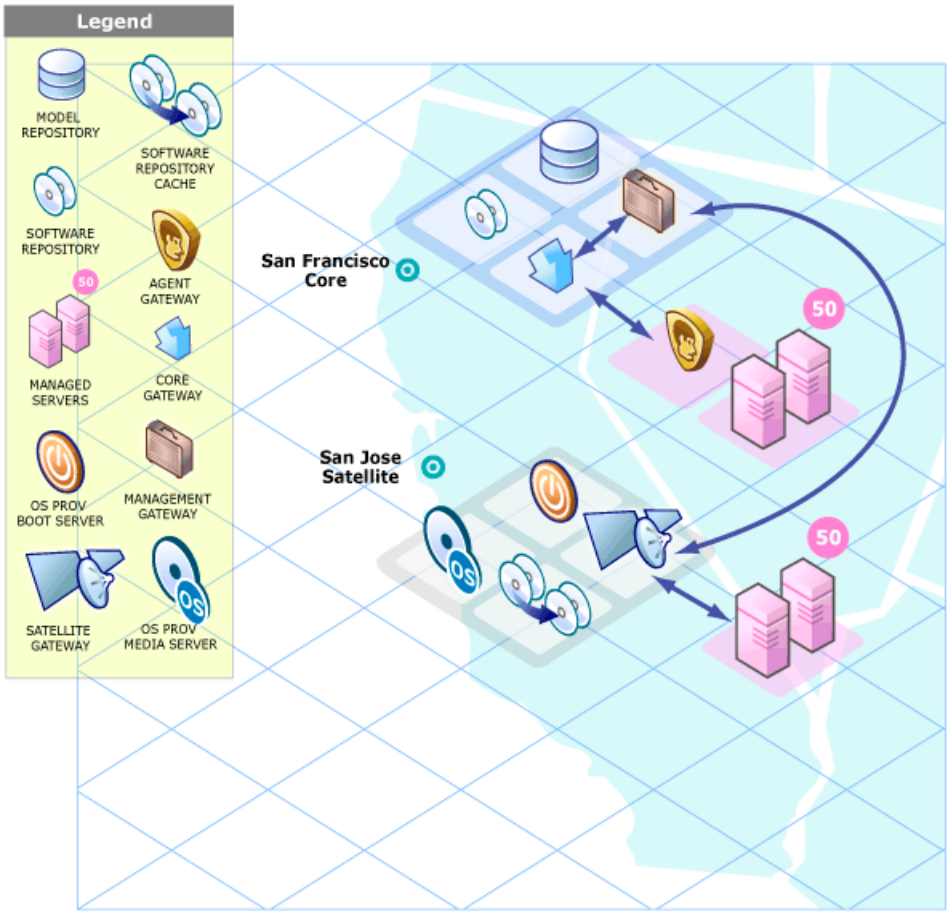
A Simple Single Core to Satellite Link

Figure 12 shows a single Satellite linked to a Single Core. In this example, the main facility is in San Francisco, and a smaller remote facility is in San Jose.

The San Francisco Single Core consists of several components, including the Software Repository, the Model Repository, an Agent gateway and a Management Gateway. For simplicity, this figure does not show all required Core Components, such as the Command Engine.

The San Jose Satellite consists of a Software Repository Cache, an Satellite Gateway, and an optional OS Provisioning Boot server and Media Server.

Figure 12 Satellite with the Single Core



For a more detailed description of these SA components, see Software Repository Cache on page 31, Boot Server on page 31, and Media Server on page 31.

The San Jose Satellite's Software Repository Cache contains local copies of software packages to be installed on Managed Servers in that facility.

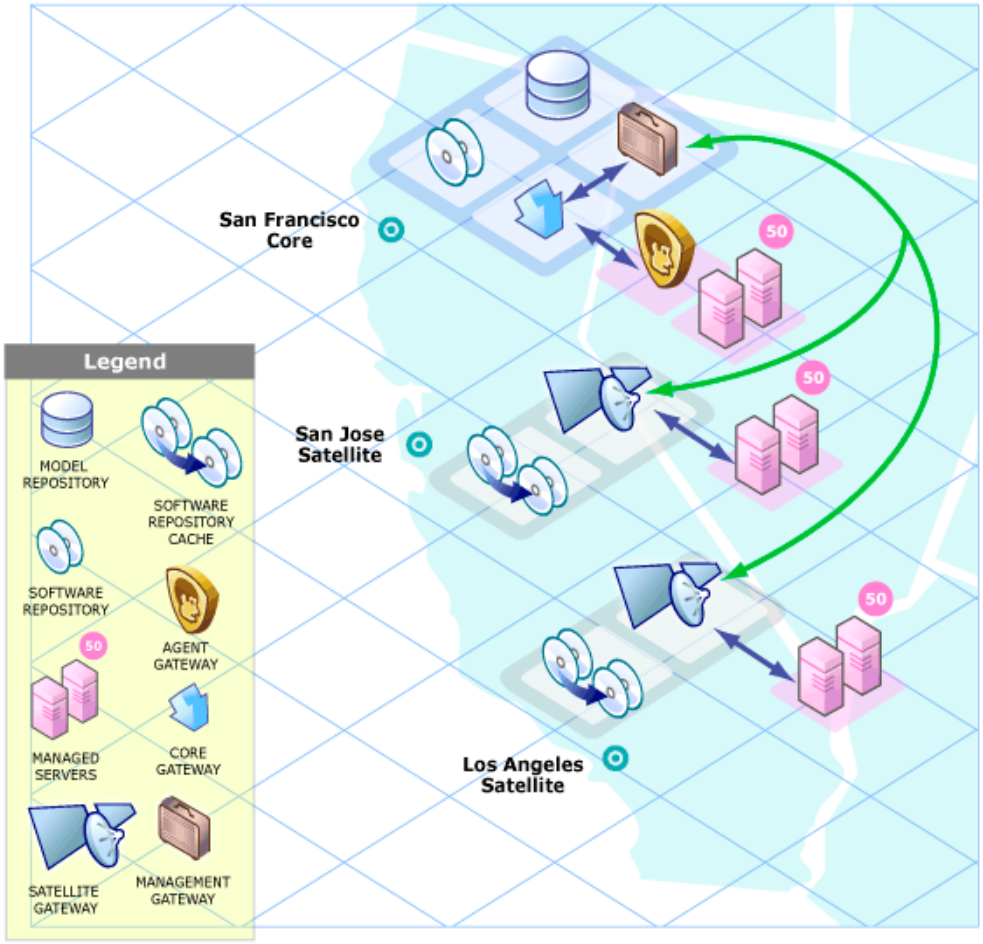
The Server Agents installed on managed servers at the San Jose facility connect to the San Francisco core through the San Jose Satellite Gateway which communicates with the San Francisco Management Gateway, then through the San Francisco Core gateway, ultimately, with the required Core Components.

Return communication reverses that path. The Server Agents installed on managed servers in the San Francisco facility communicate with the Core Components through the San Francisco facility's Agent and Core Gateways.

A Two Satellite to Single Core Link

Figure 13 shows two Satellites linked to a Single Core. In this example, San Francisco is the main facility, Sunnyvale and San Jose are Satellite facilities.

Figure 13 Two Satellites with a Single Core

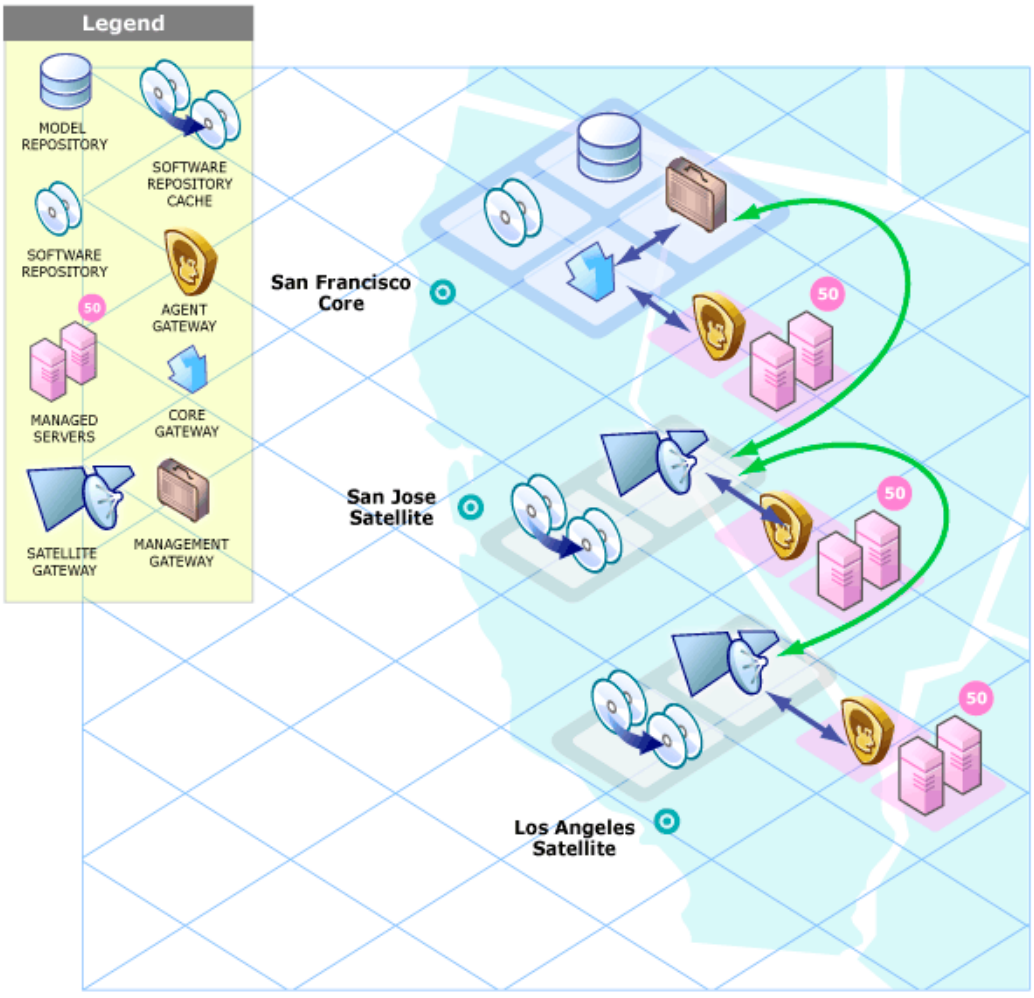


A Cascading Satellite Link

Figure 14 shows cascading Satellites, a topology in which Satellite Gateways are connected in a *chain*. This topology enables you to create a hierarchy of Software Repository Caches. Note that, the Satellite Gateways in this topology must belong to different SA Realms.

When tasked to install a package on a managed server in the Sunnyvale facility, SA first checks to see if the package resides in the Software Repository Cache in Sunnyvale. If the package is not in Sunnyvale, then SA checks the Software Repository Cache in San Jose. Finally, if the package is not in San Jose, SA goes to the Software Repository in the San Francisco core. For more information, see “Managing the Software Repository Cache” in the *SA Administration Guide*.

Figure 14 Cascading Satellites with a Single Core



Satellites in a Multimaster Mesh

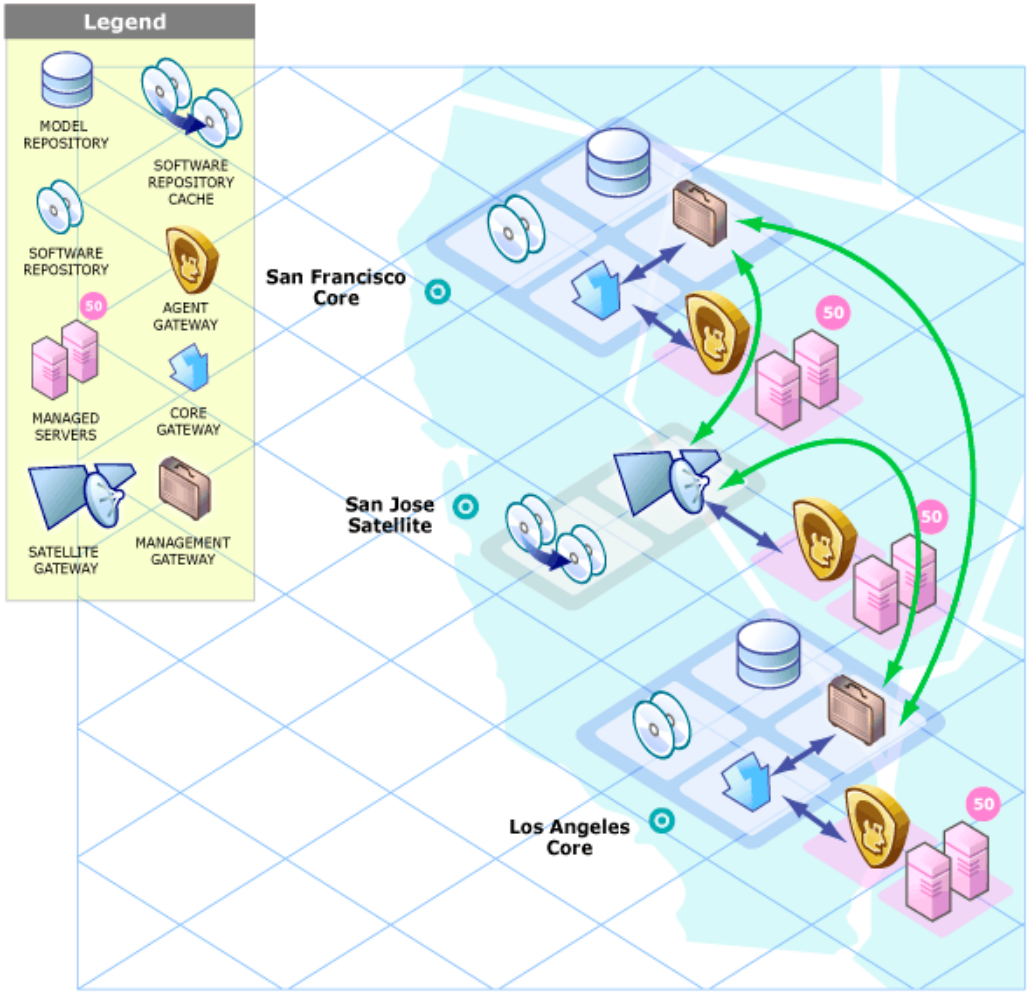
Figure 15 shows the San Jose Satellite connected to two SA Cores in a Multimaster Mesh.

Even when communication is possible to both Los Angeles and San Francisco, the Management Gateway chooses the route with the lowest cost (in Figure 15, the San Francisco route). You control cost evaluation using a parameter specified during Gateway installation. System designers can specify rules governing which SA Gateway routes to use to minimize network connectivity costs.

Using the same example environment in a failover scenario, during normal operations, the servers in the San Jose Satellite are managed by the San Francisco Core. Note, however, that the San Francisco and the Los Angeles Cores are directly connected through their Management Gateways.

If the connection between the San Jose Satellite and the San Francisco Core fails, the San Jose Satellite Gateway can immediately move communications from San Francisco to the Los Angeles core, allowing that core to maintain management of the San Jose servers. The Los Angeles Core will have up-to-date information about the San Jose site because the San Francisco Core's Model Repository data will have been replicated to the Los Angeles Model Repository as a part of normal SA operations.

Figure 15 Satellite in a Multimaster Mesh



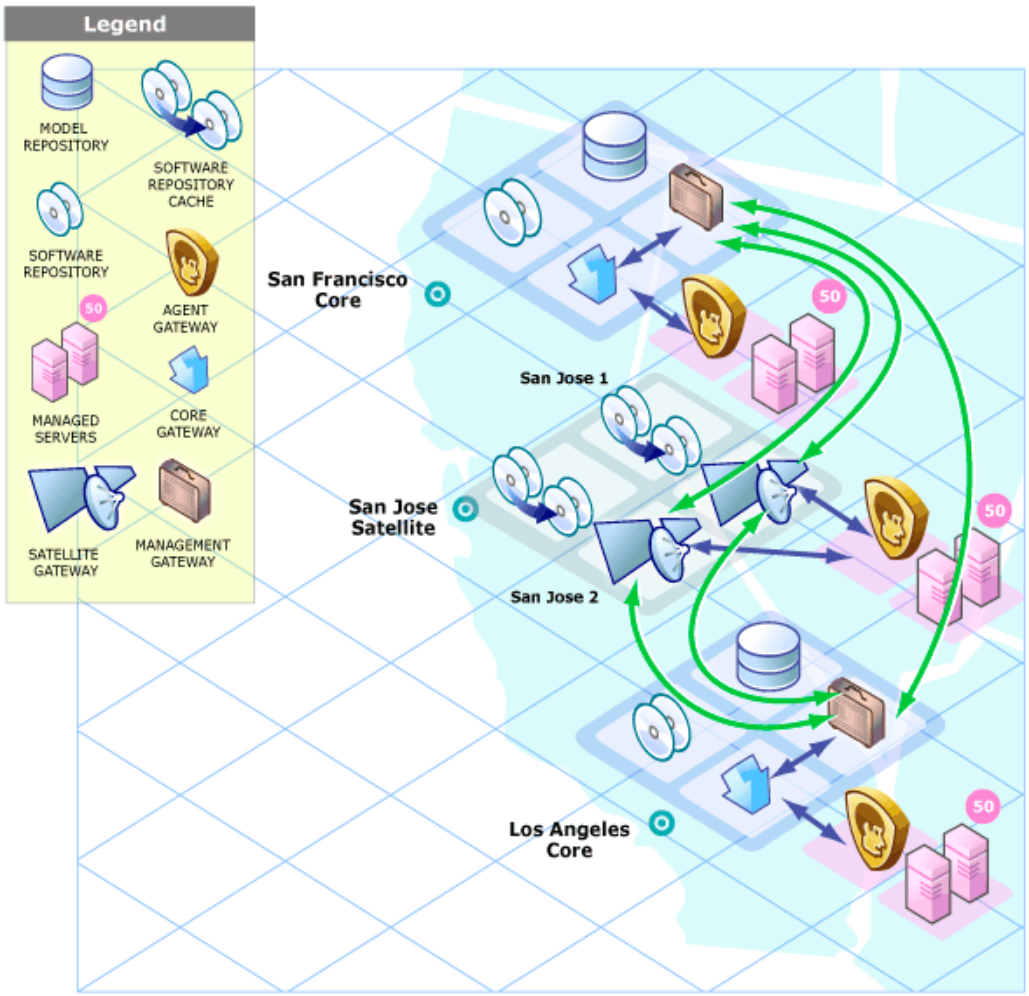
Satellite With Multiple Gateways in a Multimaster Mesh

Figure 16 shows a topology that provides failover capability in two ways. First, the San Jose Satellites 1 and 2 have Gateway connections to both the San Francisco and Los Angeles Management Gateways. If the Los Angeles core becomes unavailable, the San Francisco core can still manage the servers in the San Jose Satellite.

Second, the Agents installed on the Managed Servers in the San Jose Facility point to both of the Satellite's Agent Gateways. SA Agents automatically load balance over the available Agent Gateways and therefore can communicate directly with either the San Francisco or Los Angeles cores.

If one Gateway becomes unavailable, the Agents that are using the unavailable gateway as their primary gateway will automatically failover to using the secondary gateway. During routine agent-to-core communication, SA Agents will discover new gateways added to (or removed from) the Satellite.

Figure 16 Satellite With Multiple Gateways in a Multimaster Mesh

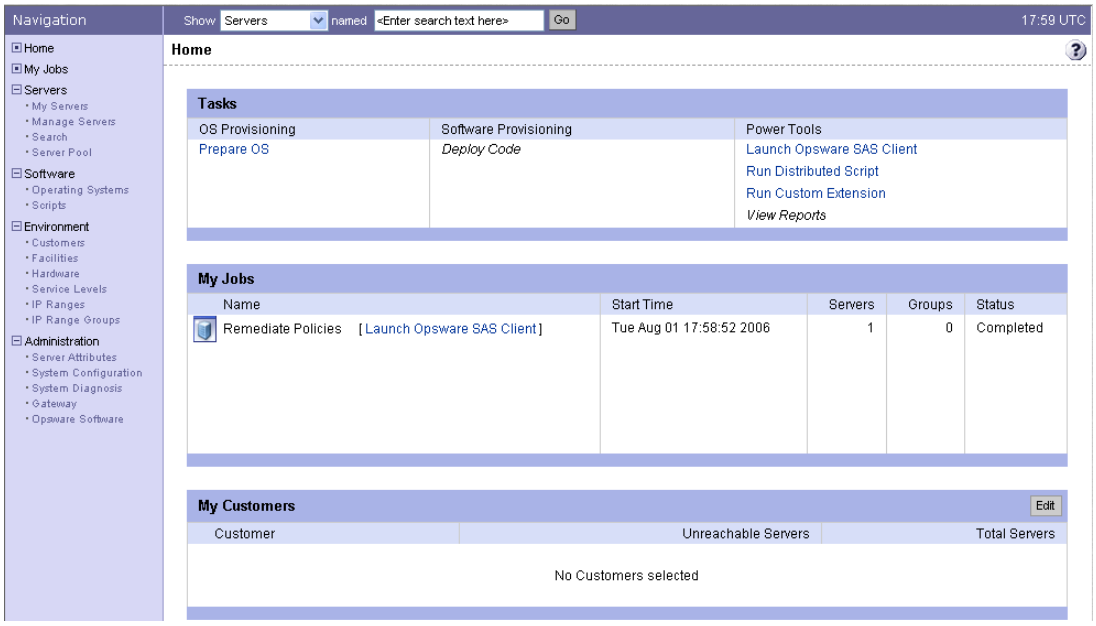


SA Interfaces and Tools

Depending on the type of operation you need to perform with SA, you select the appropriate user interface.

- **SAS Web Client**: A browser-based user interface to SA through which you:
  - Manage servers
  - Provision applications and operation systems onto servers
  - Run distributed scripts on servers
  - Deploy code and content to servers, among other things.

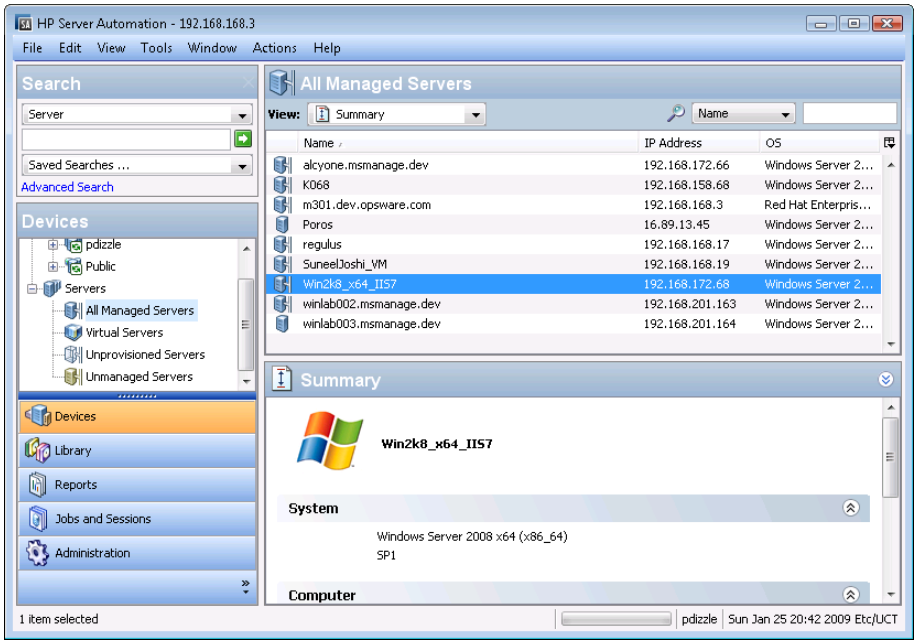
Figure 17 The SAS Web Client Main Page



- **SA Client**: A Java Web-Start application that extends the SAS Web Client features and provides the following new features:
  - Unmanaged Server Discovery and Agent Deployment Utility
  - Device Explorer
  - Virtual Server Management
  - Server Automation Visualizer (SAV)
  - Audit and Remediation
  - Compliance View
  - Reports
  - Software Management
  - Script Execution
  - Patch Management for Windows
  - Patch Management for Unix
  - Patch Management for Solaris

- Application Configuration Management
- Global Shell
- Network Automation (NA) Integration

Figure 18 The SA Client Main Page



- **Command Line Interface (OCLI)**: A command-line interface that you can use to access many SA functions to perform bulk operations or repetitive tasks on multiple servers. In SA, the command-line environment consists of the Global Shell (OGSH), the Global File System (OGFS), and Command-line Interface (OCLI) methods.
- **DCML Exchange Tool (DET)**: A utility that exports almost all server management content from one SA Core and imports it into another core. SA also provides pre-packaged server management content appropriate for new installations that can be imported into an SA Core using DET after initial setup. See the *SA Content Utilities Guide* for information about using this utility.
- **Intelligent Software Module Development Kit (IDK)**: A development kit that allows administrators to deliver stable and consistent software builds and manage change in complex data center environments. The IDK consists of command-line tools and libraries for creating, building, and uploading intelligent software modules (ISMs). An ISM is a set of files and directories that include packages, installation scripts, control scripts, and so on. See the *SA Content Utilities Guide* for information about using the IDK Development Kit.
- **SA APIs**: A set of APIs and a command-line interface (CLI) that facilitate the integration and extension of SA. This platform allows other IT systems — such as existing monitoring, trouble ticketing, billing, and virtualization technology — to exchange information with SA. This broadens the scope of how IT can use SA to achieve operational goals.
- **Automation Platform Extensions (APXs)**: APXs provide a framework that allows anyone familiar with script-based programming tools such as shell scripts, Python, Perl, and PHP, to extend the functionality of SA and create applications that are tightly integrated into SA. SA provides two types of APXs:

— **Program APXs** (also called *Script APXs*) run in the Global File System (OGFS) and can use all of the OGFS functionality

— **Web APXs**: allow you to create a web-based application, where either an Apache 2.x process or a CGI/PHP script is called using GET or POST URL.

For more information about APXs, see the *SA Platform Developers Guide*.

SA Product Options

SA provides a set of options that enable automation of many IT processes, including:

- OS Provisioning
- Software Management
- Application Configuration Management
- Patch Management for Windows
- Patch Management for Unix
- Patch Management for Solaris
- Audit and Remediation
- Virtual Server Management
- Server Automation Visualizer (SAV)
- Storage Visibility and Automation
- Code Deployment & Rollback
- Reports
- Configuration Tracking

All SA options support cross-platform environments and are designed to automate both new and existing data center environments.

OS Provisioning

OS Provisioning provides you with the ability to install (or *provision*) pre-configured operating systems on servers in your facility, ensuring that each server in your facility has a standardized, default operating system configuration that you control. For detailed information about OS Provisioning, see the *SA Policy Setters Guide* and the *SA User Guide: Application Automation*.

SA OS Provisioning supports:

- Windows, Solaris, and Linux.
- Network or CD/DVD-based installations.
- Separation of duties between data center staff and systems administrators.
- A model-based approach — in which you create a *standard build* in SA which can then be installed on many systems.

OS Provisioning integrates with your operating system vendors' native installation technology, specifically:

- Windows setup answer files: `unattend.txt, unattend.xml, sysprep.inf`
- Red Hat Kickstart
- SuSE YaST (Yet another Setup Tool)
- Solaris Jumpstart
- WINPE/WIN-BCOM/UNDI

You can provision an operating system on:

- A server in SA's unmanaged server pool that does not have an operating system installed (*bare metal sever*)
- A server in SA's *unmanaged server pool* with an installed operating system
- A server in SA's *managed server pool* with an installed operating system (*reprovisioning*)

You can perform OS Provisioning functions from both the SA Client and the SAS Web Client.

SA automates the following operating system installation tasks:

- Preparing hardware for operating system installation using configuration specifications contained in an OS Installation Profile.
- Defining OS Build Plans which are a list of tasks to be performed on a server before and after operating system installation. OS Build Plans are more powerful than and can be used in place of OS Sequences.
- Defining OS Sequences which are a list of tasks to be performed on a server during installation. OS Sequences can include application, patch, and remediation policies. SA recommends that you use the more flexible OS Build Plans.
- Installing a baseline operating system and default operating system configuration.
- Applying the latest set of operating system patches after the operating system has been installed.
- Installing system agents and utilities such as SSH, PC Anywhere, backup agents, monitoring agents, or anti-virus software.
- Installing widely-shared system software such as Java Virtual Machines.

Software Management

The Software Management feature in HP Server Automation provides a powerful mechanism to model software by using software policies and to automate the process of deploying software and configuring applications on a server in a single step. In addition, the Software Management feature provides a structure to organize your software resources in folders and define security permissions around them. This feature allows you to verify the compliance status of a server and remediate non-compliant servers.

The Software Management feature in SA Client provides the following functions:

- Create an organizational structure for software
- Define security boundaries for folders
- Define a model-based approach to manage the IT environment in your organization
- Enable sharing of software resources among user groups
- Deploy and configure applications simultaneously
- Deploy multiple application instances on one server

- Establish a software deployment process
- Verify compliance status of servers to software policies
- Generate reports
- Comprehensively search for software resources and servers

Application Configuration Management

Application Configuration Management (ACM) allows you to create templates so you can modify and manage application configurations associated with server applications. ACM enables you to manage, update, and modify those configurations from a central location, ensuring that applications in your facility are accurately and consistently configured.

Using ACM, you can perform the following tasks:

- Manage configurations based on files and objects, such as Windows registry, IIS metabase, WebSphere, COM+, and more.
- Preview configuration changes before applying them.
- Edit and push configuration changes to individual servers or server groups.
- Use information in the SA data model to set configuration values.
- Manage configurations of any application by building configuration templates.
- Audit the Application Configurations on a server to determine if any of the configuration files on the server are out of sync with the values stored in your templates.

Patch Management for Windows

The Patch Management for Windows feature enables you to identify, install, and remove Microsoft® Windows patches and maintain a high level of security across managed servers in your organization. With SA Client user interface, you can identify and install patches that support security vulnerabilities for the Windows 2000, Windows 2003, and Windows NT4.0 operating systems. These patches include Service Packs, Update Rollups, and hotfixes.

Patch Management for Unix

The Patch Management for Unix feature enables you to identify, install, and remove Unix patches to maintain a high level of security across managed servers in your organization. With the SA Client, you can identify and install patches that support security vulnerabilities for the AIX, HP-UX, Linux, and Solaris operating systems.

Patch Management for Solaris

HP Server Automation patch management for Solaris allows you to automate the process of installing and uninstalling Solaris patches and patch clusters on Sun Solaris using patch policies. In addition, SA analyzes the dependency, supersedence, and applicability relationships between patches in the policy and displays an updated and ordered list of patches that should be installed on the server. This feature allows you to verify the compliance status of a server and remediate non-compliant servers and automatically download the Solaris patches into SA and organize them into patch policies.

Audit and Remediation

The Audit and Remediation feature allows you to define server configuration policies and make sure that servers in your facilities meet those policy standards. When servers are found to be 'out of compliance' (not configured the way you want them to be), you can remediate the differing server configurations.

With Audit and Remediation, you can audit a server configuration values based upon a live server (or server snapshot), or based upon your own custom values, perform server comparisons against a baseline, and create custom audit policies that define company or industry server configuration compliance standards, and which can be used inside of audits, snapshot specifications, and audit policies.

Using Audit and Remediation, you can perform the following tasks:

- Compare servers or snapshots to reference servers or snapshots
- Create audits for repeated use
- Create audit policies that define compliance and security standards for your organization
- Associate audits with individual servers or dynamic server groups
- Remediate problems at multiple levels, including files, directories, patches, registry keys, and packages

Virtual Server Management

SA Virtual Server Management enables you to provision and manage virtual servers for Solaris 10 local zones, VMware ESX 3, 3.5 3.0.2, and ESXi, as well as, Microsoft Hyper-V virtual machines (VMs).

Using the SA Client, you can perform the following tasks:

- View both hypervisor and virtual servers and their relationships.
- View virtual servers and their relationship in HP Server Automation Visualizer (SAV).
- Provision VMware ESX and Solaris 10 hypervisors on bare metal servers.
- Provision VMware virtual machines (VM) using OS Provisioning.
- Create, start, stop, modify, and remove Solaris local zones.
- Deploy agents on unmanaged virtual servers using the Agent Discovery and Deployment for VMware ESX VMs.
- Search for virtual servers in your data center using the Search tool.
- Create dynamic Device Groups based upon virtual server characteristics (zones or VMs).

Server Automation Visualizer (SAV)

The Server Automation Visualizer (SAV) feature is designed to help you optimally understand and manage the operational architecture and behavior of distributed business applications in your IT environment. Since these applications are complex collections of services that typically run across many servers, as well as network and storage devices, it can become increasingly difficult to understand (or remember) what is connected to what, where performance problems originate, how to troubleshoot and resolve problems, and what result would occur if you make a change in your environment.

SAV helps you see (visualize) this type of information through physical and logical drawings.

Storage Visibility and Automation

The Storage Visibility and Automation feature offers storage management capabilities by enabling end-to-end visibility and management of the entire storage supply chain. This feature helps server administrators day-to-day tasks by providing tools that increase cost savings through application storage, dependency and visibility, storage audits, storage capacity and utilization trending, and scripting and automation. See the Storage Visibility and Automation User's Guide for more information.

Code Deployment & Rollback

SA automates code and content deployment to reduce the risk and time requirements associated with pushing new code to production. The Code Deployment & Rollback (CDR) feature provides an automated system for deploying code (such as, ASP, JSP, JAR, Java, C++, and Perl files) and content (such as, HTML, JPEG, GIF, and PDF files).

Specifically, CDR enables you to perform the following actions:

- Push code from staging or development environments to production environments.
- Synchronize code and content across multiple servers and locations.
- Automatically rollback to the previous version of code or content.
- Sequence multiple, complex deployment steps into repeatable workflows.
- Manage changes across heterogeneous operating systems.

Reports

The Reports feature provides comprehensive, real-time information about managed servers, network devices, software, patches, customers, facilities, operating systems, compliance policies, and users and security in your environment. These reports are presented in graphical and tabular format, and are actionable—where you can perform appropriate actions on objects, such as a policy or an audit, within the report. These reports are also exportable to your local file system (in .html and .xls formats) to facilitate use within your organization.

Configuration Tracking

The Configuration Tracking feature tracks, backs up, and recovers critical software and system configuration information across Unix and Windows servers.

System administrators set up policies that describe the configuration files and databases to track, and the actions to take when a change in configuration is detected. Policies can be assigned to software, individual servers, groups of servers, and customers, and applied either locally or globally across data centers.

When SA notices a server configuration change, it can log the change, notify administrators about the change with email, or back up the configuration, depending on the policy set by the administrator.

When a bad configuration change forces administrators to rollback to a previous version, they can use SA to restore the configuration file to the saved version of the configuration. By notifying users about configuration changes — and maintaining a version history of those changes — organizations can quickly diagnose problems related to configuration errors and rollback to a known good state. In addition, this capability helps teams plug security holes inadvertently created by bad server configurations.

Typically, system administrators define configuration-tracking policies on a per-application basis. So for example, a policy for BEA WebLogic might specify, "Monitor the `weblogic.conf` file, notify `app-server-admins@company.com` of any changes, and maintain a version history of any changes that occur for 30 days." After a policy is defined in this fashion, administrators can apply the policy to all the WebLogic servers running in their environment or to specific servers.

SA Utilities

SA also provides utilities that assist with typical data center automation tasks.

Script Execution

The Script Execution feature enables you to share and run ad-hoc or saved scripts across an entire farm of SA-managed servers.

Executing scripts with SA instead of manually, has the following advantages:

- Parallel script execution across many Unix and/or Windows servers, saving time and ensuring consistency.
- Role-based access control, ensuring only authorized administrators can execute scripts on hosts to which they have access.
- The ability to control access to scripts by storing them in private or in public libraries.
- The ability to see and download script output one server at a time or in a consolidated report, which captures output from all servers in a single place.
- The ability for scripts to be mass-customized. Administrators can access information in SA about the environment and the state of servers. This is critical to ensuring that the right scripts are executed on the right servers.
- A comprehensive audit trail that reports who, what, when, and where a particular script was executed.
- Using known system state and configuration information to customize script execution, you can tailor each script by referencing and accessing the information in SA, such as the customer or business that owns the server, whether the server is a staging or production server, which facility the server is located in, and custom name-value pairs.
- You can share scripts with others without compromising security because SA maintains strict controls on who can execute scripts on which servers and generates a comprehensive audit trail of script execution.

Unmanaged Server Discovery and Agent Deployment Utility

The SA Discovery and Agent Deployment (ODAD) utility allows you to deploy Server Agents to a large number of servers in your facility and place them under SA management.

Using ODAD, you ca:

- Scan your network for unmanaged servers.
- Select unmanaged servers for SA Server Agent installation and deploy the agent(s).
- Select a communication tool and provide user/password combinations.
- Choose Server Agent installation options.

Device Explorer

The Device Explorer lets you view information about servers in your SA managed environment. The Device Explorer consists of:

The *Server Explorer*:

- Create a server snapshot, perform a server audit, audit application configurations, create a package, and open a remote terminal session on a remote server.
- Browse a server's file system, registry, hardware inventory, software and patch lists, and services.
- Browse SA information such as properties, configurable applications, and even server history.

The *Groups Browser*:

- Audit system information, take a server snapshot, and configure applications.
- View and access group members (servers and other groups).
- View group summary and history information.

Compliance View

The Compliance Dashboard allows you to view the overall compliance levels for all the devices in your facility and helps you to remediate compliance problems. The Compliance Dashboard displays compliance tests for software policies, application configurations, audits, patches, and duplex status. Each of these compliance tests is based upon an SA policy (user- or system-defined) which define a unique set of server or device configuration settings or values that ensure that your IT environment is configured the way you want it to be.

Global Shell

The Global Shell enables you to manage servers by using a command-line interface. You can remotely perform the following tasks:

- Complete routine maintenance tasks on managed servers.
- Troubleshoot, identify, and remediate problems on managed servers.

Global Shell consists of a file system and a command-line interface to that file system for managing servers in SA. The file system is known as the SA Global File System (OGFS). All object types in the OGFS (such as servers, customers, and facilities) are represented as directory structures in this file system.

The Global Shell feature also manages user permissions for accessing the file system, Windows Registry, and Windows Services objects on managed servers.

Network Automation (NA) Integration

Network Automation (NA) Integration enables you to closely examine detailed information about managed servers and the network devices connected to them so that you can determine how they are related and then, subsequently, coordinate and implement those changes. This feature supports an integrated approach to using NA and SA so that you can perform actions on device groups, such as combine event history, determine compliance, and identify duplex mismatches across servers and network devices in your environment.

Index

S

W