Opsware® SAS 6.5 Planning and Installation Guide

Corporate Headquarters

599 North Mathilda Avenue Sunnyvale, California 94085 U.S.A.
T + 1 408.744.7300 F +1 408.744.7383 www.opsware.com

Opsware SAS Version 6.5.1

Opsware is protected by U.S. Patent Nos. 6,658,426, 6,751,702, 6,816,897, 6,763,361 and patents pending.

Additional proprietary information about third party and open source materials can be found at http://www.opsware.com/support/sas651tpos.pdf.

Table of Contents

Preface

Welcome to the Opsware Server Automation System (SAS) – an enterprise-class software solution that enables customers to get all the benefits of the Opsware data center automation platform and support services. Opsware SAS provides a core foundation for automating formerly manual tasks associated with the deployment, support, and growth of server and server application infrastructure.

Overview of this Guide

This guide describes how to use the Opsware Installer to install the software components that make up an Opsware core. It also describes the administrative tasks required prior to installing an Opsware core.

This guide is intended for Unix system administrators, database administrators, and network administrators.

Contents of this Guide

This guide contains the following chapters:

Chapter 1: Opsware SAS Architecture: Provides an overview of Opsware SAS architecture, includes information needed for Opsware core or Opsware Satellite installation, and presents the different topologies of Opsware SAS.

Chapter 2: Operating Systems and Hardware Requirements: Describes the supported operating systems for an Opsware SAS core, managed servers, and the SAS Client. This chapter also describes the hardware requirements for the servers running an Opsware SAS core and provides guidelines on how to distribute Opsware SAS components across the servers running an Opsware SAS core.

Chapter 3: Pre-installation Requirements: Describes the system and network administration tasks that must be performed before you can run the Opsware Installer.

Chapter 4: Installation Methods and Checklists: Describes the types of Opsware SAS installation, reviews the Opsware SAS core installation process, and provides checklists to aid you in gathering required information for Opsware SAS core installation.

Chapter 5: Prerequisites for the Installer Interviewer: Lists the information needed to complete the Opsware Installer interviewer and provides information about installer command line syntax, log files, and Opsware Installer distribution on DVDs.

Chapter 6: Standalone Core Installation: Describes how to run the Opsware Installer to create a standalone core.

Chapter 7: Standalone Core Post-Installation Tasks: Describes system administration tasks that you must perform after installing a core.

Chapter 8: Multimaster Installation: Describes how to run the Opsware Installer to upgrade a standalone core to multimaster and install target facilities.

Chapter 9: Satellite Installation: Describes how to run the Opsware Installer for creating an Opsware satellite realm.

Chapter 10: Opsware SAS Configuration: Provides an overview of the configuration tasks required for Opsware SAS after the core has been installed.

Chapter 11: Opsware Core Uninstallation: Shows how to uninstall a standalone core, remove a core from a multimaster mesh, and uninstall an entire Opsware SAS made up of multiple cores in different facilities.

Appendix A: Oracle Setup for the Model Repository: Explains how to configure and maintain your Oracle database to work with the Model Repository.

Appendix B: TIBCO Rendezvous Configuration for Multimaster: Provides reference information about the TIBCO configuration for multimaster. By default, the Opsware SAS Installer configures TIBCO for a multimaster mesh.

Appendix C: Opsware Gateway Properties File: Provides reference information about the settings in the properties file used by the Opsware Gateway.Conventions in this Guide

This guide uses the following typographical and formatting conventions.

NOTATION	DESCRIPTION
Bold	Identifies field menu names, menu items, button names, and inline terms that begin with a bullet.

NOTATION	DESCRIPTION
`Courier`	Identifies text that is entered or displayed at the command-line prompt, such as Unix commands, Opsware SAS commands, file names, paths, directories, environment variable names, contents of text files that are viewed or edited with a text editor, source code in a programming language, and SQL (database) commands.
Italics	Identifies document titles, DVD titles, web site addresses. Used to introduce new terms when they are first defined in a document and for emphasis.

Icons in this Guide

This guide uses the following icons.

ICON	DESCRIPTION
	This icon represents a note. It identifies especially important concepts that warrant added emphasis.
	This icon represents a requirement. It identifies a task that must be performed before an action under discussion can be performed.
	This icon represents a tip. It identifies information that can help simplify or clarify tasks.
	This icon represents a warning. It is used to identify significant information that must be read before proceeding.

Guides in the Documentation Set and Associated Users

- The *Opsware® SAS User's Guide: Server Automation* is intended for system administrators responsible for all aspects of managing servers in an operational environment. It describes how to use Opsware SAS, introducing the system and the user interface. It provides information about managing servers, remediating servers, script execution, configuration tracking, deploying and rolling back code, and agent deployment. It also explains how to use the Opsware Global Shell and open a Remote Terminal on managed servers.
- *Opsware® SAS User's Guide: Server Automation* is intended for system administrators responsible for performing the day-to-day functions of managing servers. It reviews auditing and compliance, software packaging, visual application management, application configuration, and software and operating system installation on managed servers.
- The *Opsware® SAS Administration Guide* is intended for administrators responsible for monitoring and diagnosing the health of the Opsware SAS core components. It also documents how to set up Opsware user groups and permissions.
- The *Opsware® SAS Planning and Installation Guide* is intended for advanced system administrators responsible for planning all facets of an Opsware SAS installation. It documents all the main features of Opsware SAS, scopes out the planning tasks necessary to successfully install Opsware SAS, explains how to run the Opsware Installer, and details how to configure each of the components. It also includes information on system sizing and checklists for installation.
- The *Opsware® SAS Policy Setter's Guide* is intended for system administrators responsible for setting up OS provisioning, configuration tracking, code deployment, and software management.
- The *Opsware® SAS Content Utilities Guide* is intended for advanced system administrators responsible for importing content such as software packages into Opsware SAS. It documents the following command-line utilities: OCLI 1.0, IDK, and DET (CBT).
- The *Opsware® Platform Developer's Guide* is intended for software developers responsible for customizing, extending, and integrating Opsware SAS. It documents how to create Web Services, Java RMI, Python, and CLI clients that invoke methods on the Opsware API.

Opsware, Inc. Contact Information

For more information, see the Opsware, Inc. main web site and phone number:

- *http://www.opsware.com/index.htm*
- +1 (408) 744-7300

For links to the latest product documentation and software downloads, see the Opsware Customer Support site:

- *https://download.opsware.com/opsw/main.htm*

For troubleshooting information, see the Opsware Knowledge Base:

- *https://download.opsware.com/kb/kbindex.jspa*

To contact Opsware Customer Support, see the following email address and phone number:

- support@opsware.com
- +1 (877) 677-9273

Chapter 1: Opsware SAS Architecture

IN THIS CHAPTER

This section discusses the following topics:

- Agent-Server Architecture
- Opsware SAS Topologies
- Opsware SAS Components

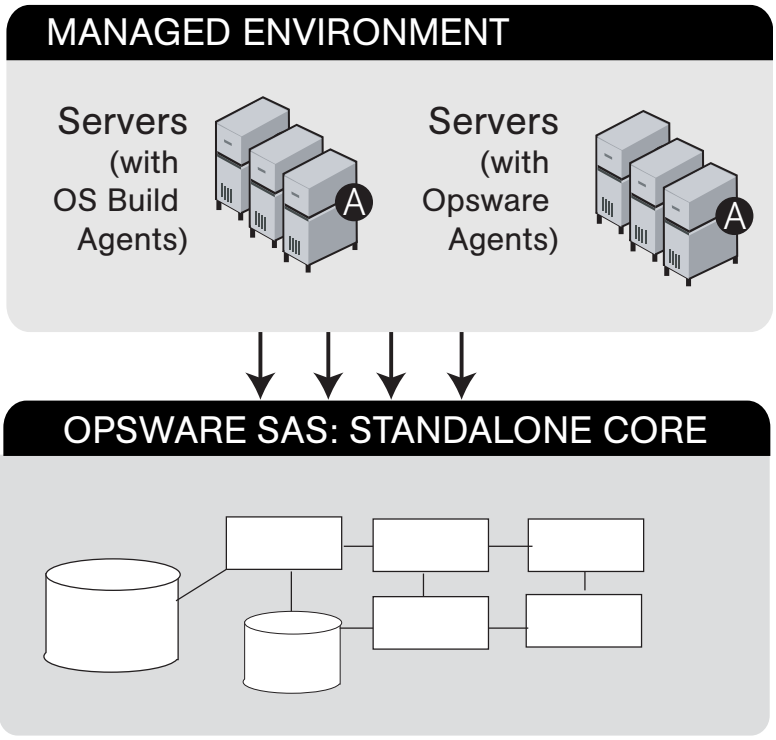
Agent-Server Architecture

This chapter provides an overview of the Opsware SAS architecture and the related information needed for installing an Opsware core or Opsware Satellite. It also describes some of the different Opsware SAS topologies to help you decide which is needed for your Opsware SAS installation.

Agent-Server Architecture of Opsware Technology

The agent-server architecture of Opsware SAS enables server management. The server portion of Opsware SAS consists of multiple, integrated components, each with a unique purpose. Each server managed by Opsware SAS runs an intelligent agent (the Opsware Agent).

Figure 1-1: Opsware SAS *Agent-Server Architecture*



The Opsware Agent is the agent of change on a server. Whenever Opsware SAS needs to make changes to servers, it does so by sending requests to the Opsware Agents. Depending on the request, the Opsware Agent on a server might use global Opsware SAS services in order to fulfill the request. For example, the Opsware Agent might often make requests to the Model Repository, the central database for all Opsware SAS components, and the Software Repository, the central repository for all software that Opsware SAS manages.

The Opsware Agent supports the following functions:

- Software installation and removal
- Configuration of software and hardware

- Periodically reporting server status
- Auditing of the server

An Opsware Agent is idle unless Opsware SAS is trying to perform some change on the server. In addition, each Opsware Agent periodically contacts the Data Access Engine and registers itself. The Data Access Engine is an XML-RPC interface to the model repository. The Data Access Engine sends this data to the Model Repository, which allows the Model Repository to keep track of server status, and know when particular servers are disconnected from or reconnected to the network.

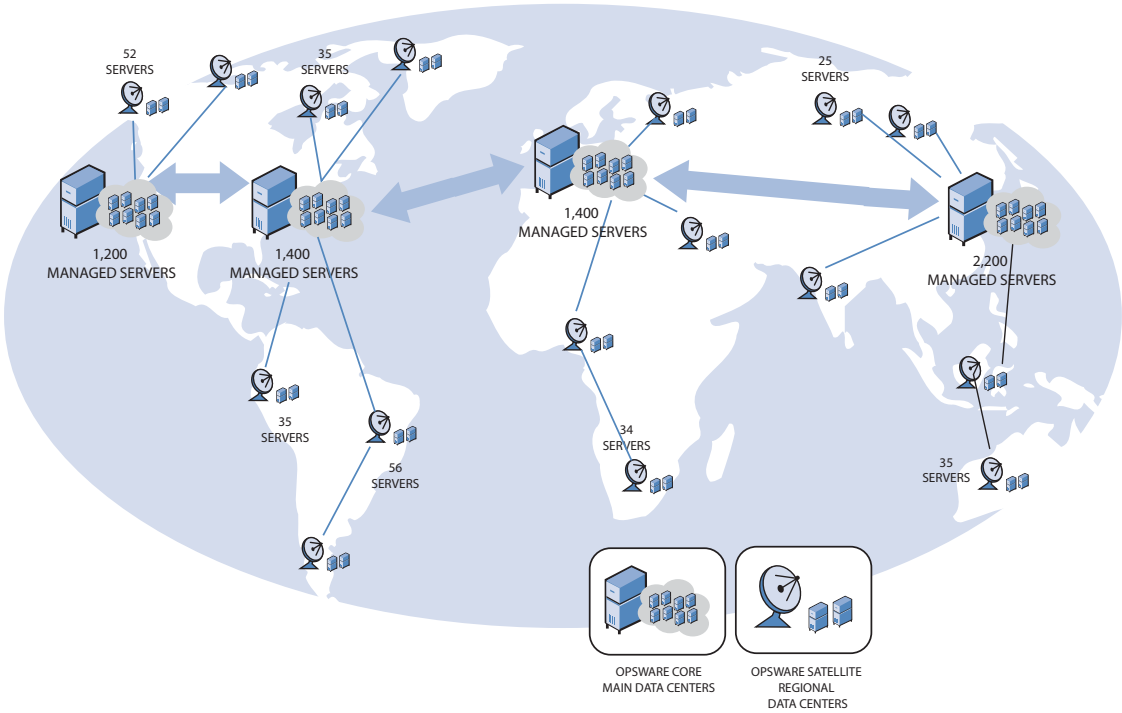
After you install an Opsware Agent on a server, users can manage the server by installing or upgrading software, patching the OS software, removing software, changing server properties, or decommissioning the server.

Server Management in Multiple Facilities

The managed environment might span several facilities. A facility refers to the collection of servers that a single Opsware Model Repository manages, and the database that stores information about the managed environment.

For example, one facility might be dedicated to an organization's Intranet, while another facility might be dedicated to the web services offered to the public.

Figure 1-2: Server Management in Multiple Facilities



Servers can be managed from any facility from a SAS Web Client or a SAS Client, regardless of which facility the server is in. For example, a user can log onto the SAS Client of a New York facility and manage servers that belong to the Los Angeles facility. When a user updates Opsware SAS data in one facility, the Model Repository for that facility is synchronized with the Model Repositories located in all remote facilities.

When using Opsware technology in multiple facilities, users should follow these work process rules to reduce the chance of data conflicts between facilities:

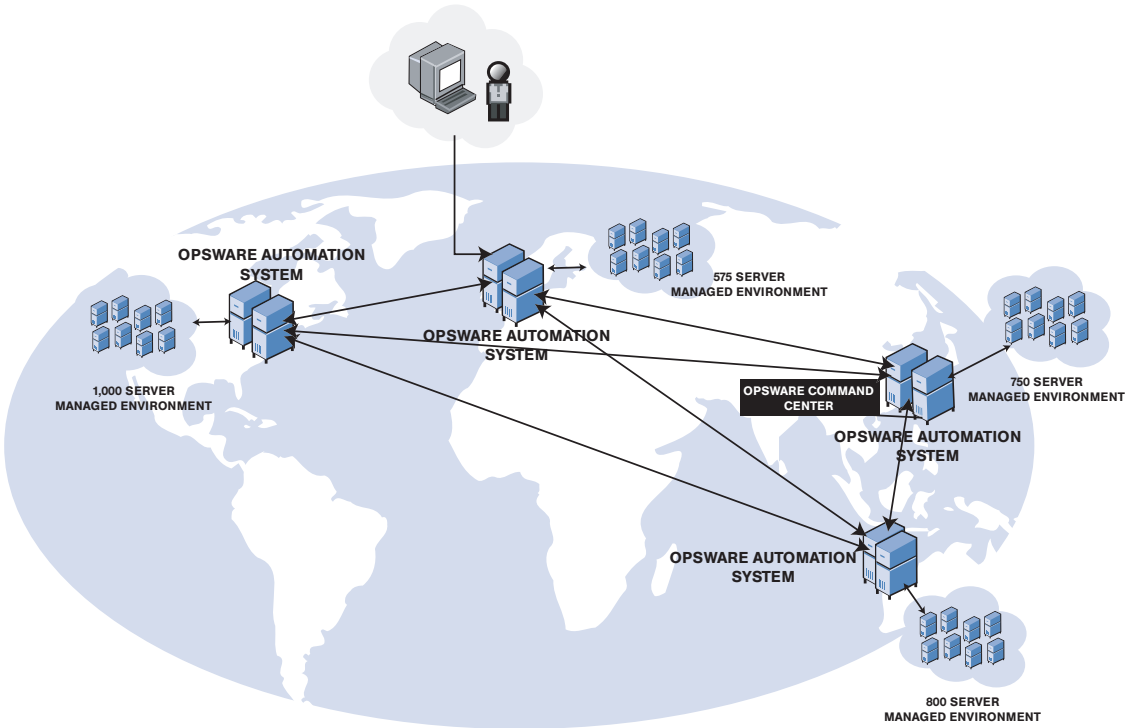
- Users should not change data in one facility and then make the same change in another facility. The changes will be automatically propagated.
- Users should not change the same object in different facilities at the same time. For example, two users should not manage the same server simultaneously from different facilities.

Multimaster Support

With the Opsware Model Repository Multimaster Component, customers can store and maintain a blueprint of software and environment characteristics for each facility so the infrastructure can be easily rebuilt in the event of a disaster. The Multimaster Replication Engine can also assist in facility migration activities and knowledge sharing across the enterprise.

Through the Model Repository Multimaster Component, Opsware SAS provides the ability to easily rebuild server and application environments, provision additional capacity, distribute updates, and share software builds, templates and dependencies – across multiple facilities and from one user interface.

Figure 1-3: Multimaster Support



A multimaster mesh is a set of two or more Opsware cores that are linked by synchronizing the data in the Model Repositories at each of the cores. The Model Repositories in each of the cores are continually updated so that they are exact duplicates of each other. All the Opsware cores in a multimaster mesh can be managed through a single SAS Web Client.

Opsware SAS Topologies

Opsware SAS requires at least one Opsware core. The simplest topology is a single, standalone core that manages servers in a single facility. To manage servers in more than one facility, you should install a multimaster mesh of cores, Opsware Satellites, or a combination of both. For more information, see the *Opsware® SAS Administration Guide*.

Benefits of Multimaster Mesh

To manage servers in large, geographically dispersed facilities, you should consider installing a core in each facility, linked in a multimaster mesh. In a multimaster mesh of cores, data is updated locally and then propagated to every Opsware Model Repository (database) in the mesh. A multimaster mesh offers the following benefits:

- **Redundancy**: Management of data is synchronized between facilities in a multimaster mesh. If the Opsware core in one facility is damaged, another core in the multimaster mesh contains a synchronized copy of the data. Also, it provides the ability to move out of a facility and keep Opsware SAS running in other facilities.
- **Performance Scalability**: An Opsware core can operate on servers in the local facility independently of the processing in the other facilities in the mesh. Only the load of the multimaster database synchronizations are transmitted between facilities.

  Write operations do not need to be proxied to a central location.
- **Geographic Scaling**: International facilities can be independent and do not need to rely on a network connection across continents to a central facility.

Example: Multimaster Topologies

Figure 1-4 shows a multimaster mesh with a core in two facilities. Each core contains a Model Repository with data that is synchronized with the other repository. This synchronization data passes through the core Gateways. The managed servers (indicated

in the figure with the letter “A”) communicate with the core via the Agent and core Gateways. If one core becomes unavailable, the managed servers in that core can still be operated on with the SAS Web Client of the other core.

See “Model Repository” on page 35 and “Opsware Gateway” on page 38 for a description of these Opsware SAS components.

Figure 1-4: Multimaster Mesh with Two Cores

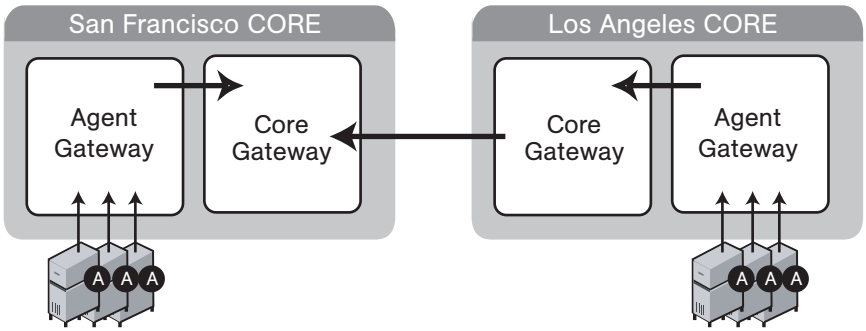
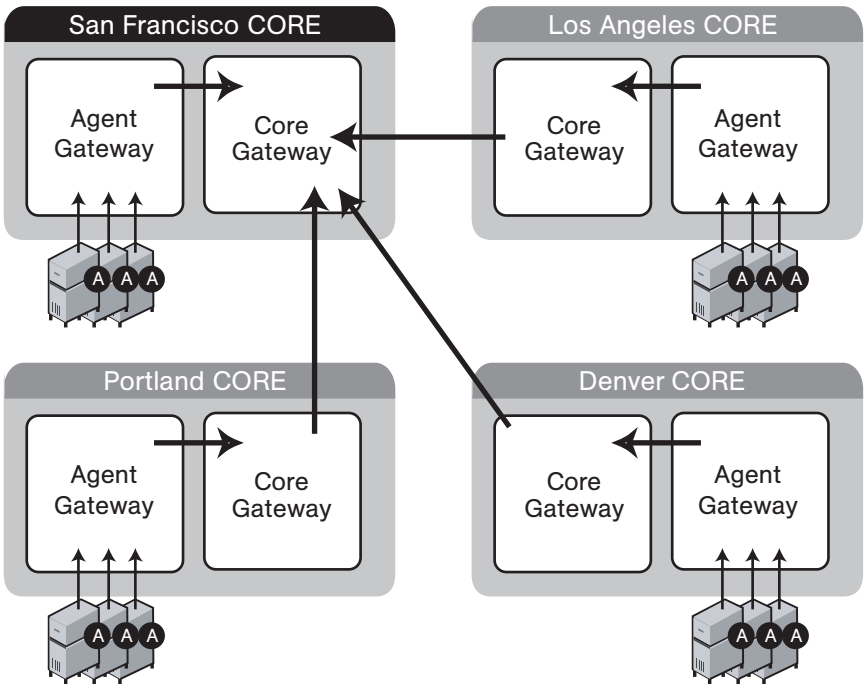


Figure 1-5 shows a multimaster mesh with several cores. This topology is in a star format with the San Francisco core at the center of the mesh. By default, the Opsware Installer configures a multimaster mesh with a star topology.

Figure 1-5: Multimaster Mesh with Four Cores



Benefits of Opsware Satellites

To manage servers in a small, remote facility, you should consider installing a Satellite in the remote facility instead of another core. Opsware Satellites offer the following benefits:

Management of servers with overlapping IP addresses: Servers in different facilities might have overlapping IP addresses. This situation can occur when servers in remote facilities are behind NAT devices or firewalls. The Opsware realm name and the IP address uniquely identify a managed server. A realm is a logical name for a group of IP addresses that can be contacted by a particular set of Gateways. Servers with overlapping IP addresses must reside in separate Opsware realms.

Network bandwidth management: Opsware SAS might share the network connection between the Satellite and the core with other applications. If this network connection has limited bandwidth, you might want to limit the network bandwidth used by Opsware SAS. You can limit the bandwidth by configuring the Opsware Gateway in the Satellite. The Opsware Gateway can manage bandwidth on a tunnel-by-tunnel basis.

Example: Satellite Topologies

Figure 1-6 shows a single Opsware Satellite linked to a standalone core. In this example, the main facility is in San Francisco, and a smaller remote facility is in San Jose. The core is made up of several components, including the Software Repository, the Model Repository, and two gateways. This figure does not show other required core components, such as the Command Engine, but indicates them with an ellipsis (...) button. When you install a standalone core, the Opsware Installer creates both the Agent and core Gateways. A Satellite can contain a Software Repository Cache, a Gateway, an OS Provisioning Boot Server, and an OS Media Boot Server.

See “Software Repository Cache” on page 38, “Boot Server” on page 34, and “Media Server” on page 34 for a description of these Opsware SAS components.

The Software Repository Cache contains local copies of software packages to be installed on managed servers in the Satellite. The Agents in the San Francisco facility communicate with the core through the Agent Gateway. The Agents in the San Jose facility connect to the San Francisco core via the Satellite Gateway.

Figure 1-6: Satellite with the Standalone Core

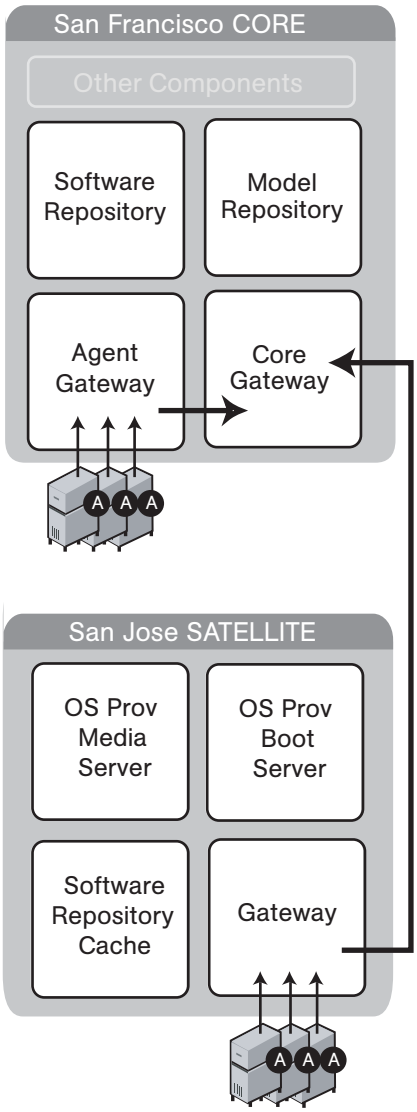


Figure 1-7 shows two Satellites linked to a standalone core. In this example, San Francisco, Sunnyvale, and San Jose are separate facilities. San Francisco is the large primary facility. Sunnyvale and San Jose are small remote facilities.

Figure 1-7: Two Satellites with a Standalone Core

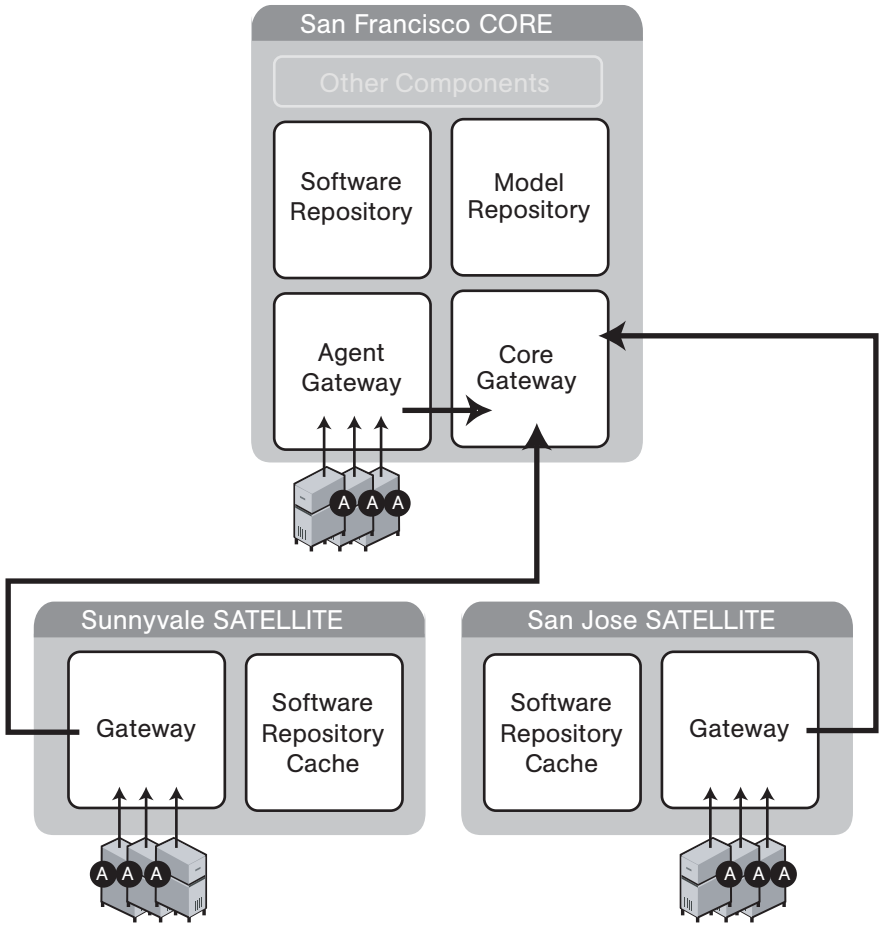


Figure 1-8 shows cascading Satellites, a topology in which Satellite Gateways are connected in a chain. This topology enables you to create a hierarchy of Software Repository Caches. The Satellite Gateways in this topology must belong to different realms. To install a package on a managed server in the Sunnyvale facility, Opsware SAS first checks to see if the package resides in the Software Repository Cache in Sunnyvale. If the package is not in Sunnyvale, then Opsware SAS checks the Software Repository

Cache in San Jose. Finally, if the package is not in San Jose, Opsware SAS goes to the Software Repository in the San Francisco core. For more information, see “Managing the Software Repository Cache” in the *Opsware® SAS Administration Guide*.

Figure 1-8: Cascading Satellites with a Standalone Core

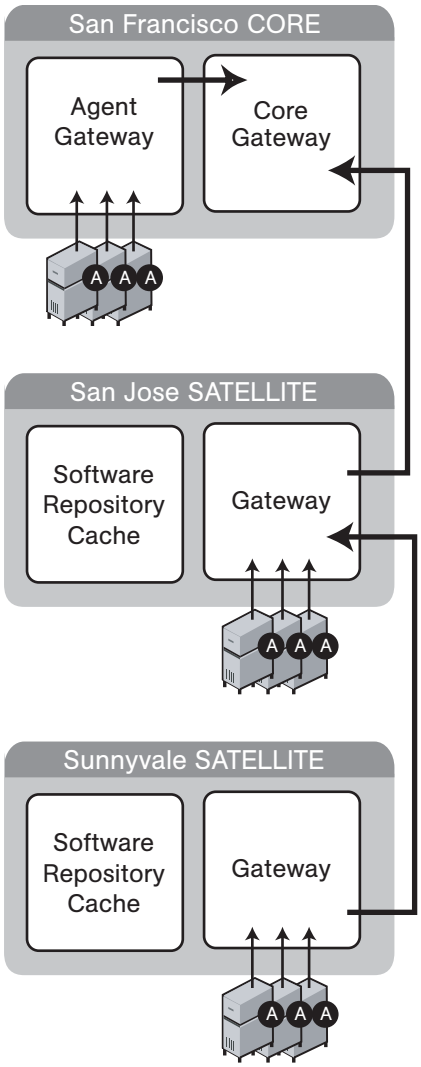


Figure 1-9 shows a Satellite connected to two cores in a multimaster mesh. A Satellite Gateway routes traffic to only one core Gateway at any given time. The Gateway chooses the route with the lowest cost, a parameter specified during Gateway installation.

Suppose that the cost of the link between the San Jose and San Francisco is the lowest. During normal operations, the servers in San Jose are managed by the San Francisco core. If the connection between San Jose and San Francisco fails, then the Gateway in San Jose will communicate with the core in Los Angeles instead.

Figure 1-9: Satellite in a Multimaster Mesh

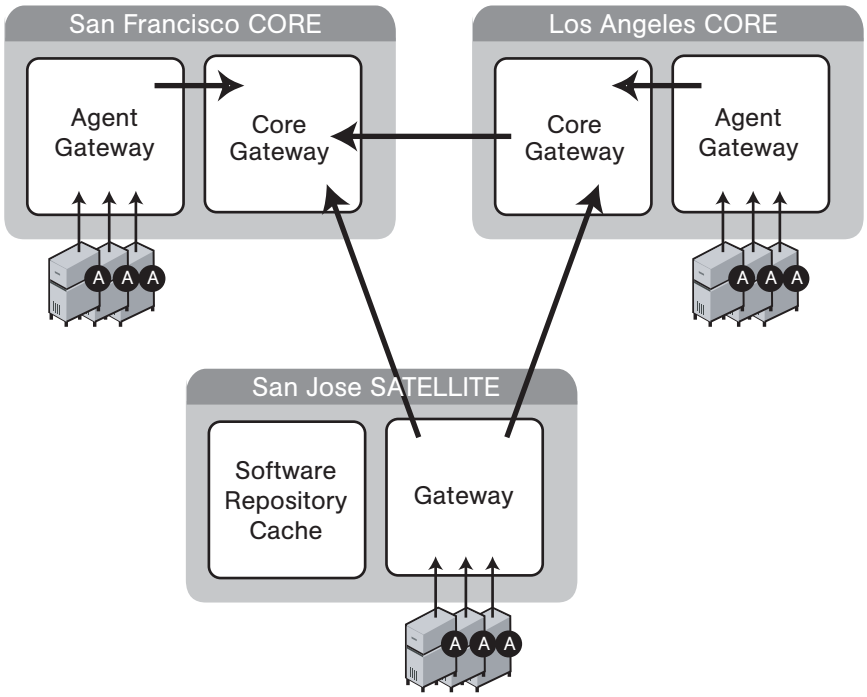
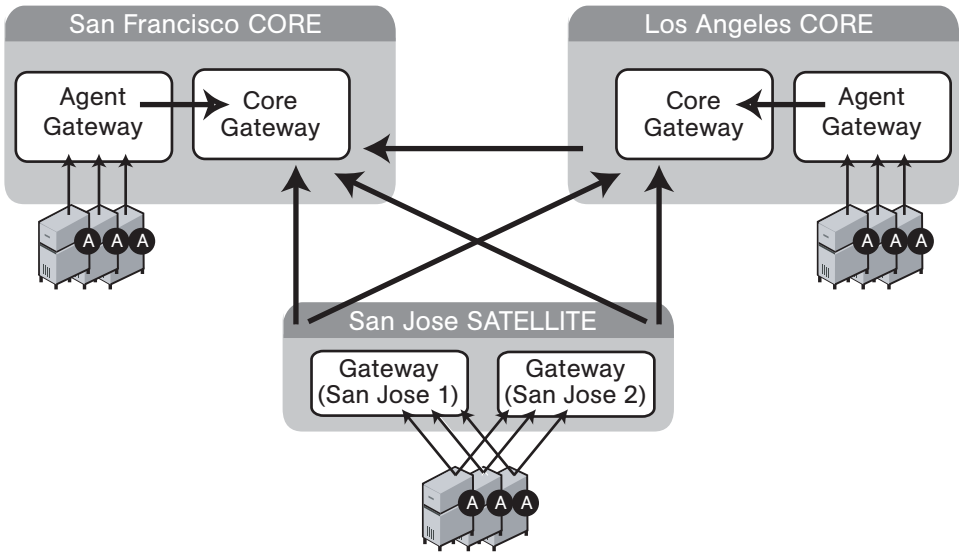


Figure 1-10 shows a topology that provides failover capability in two ways. First, the Gateway in each Satellite has connections to both core Gateways. If one core becomes unavailable, the other core can manage the servers in the Satellite. Second, the Agents in the Satellite point to both Satellite gateways. Opsware Agents automatically load balance themselves over the available gateways in a Satellite.

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, Opsware Agents will discover new gateways added to (or removed from) a multimaster mesh.

Figure 1-10: Satellite With Multiple Gateways in a Multimaster Mesh



Opsware SAS Components

Opsware SAS has an agent-server architecture. Each server managed by Opsware SAS runs an Opsware Agent, which performs tasks remotely. The server portion of Opsware SAS is called the Opsware core, consisting of multiple, integrated components, each with a unique purpose.

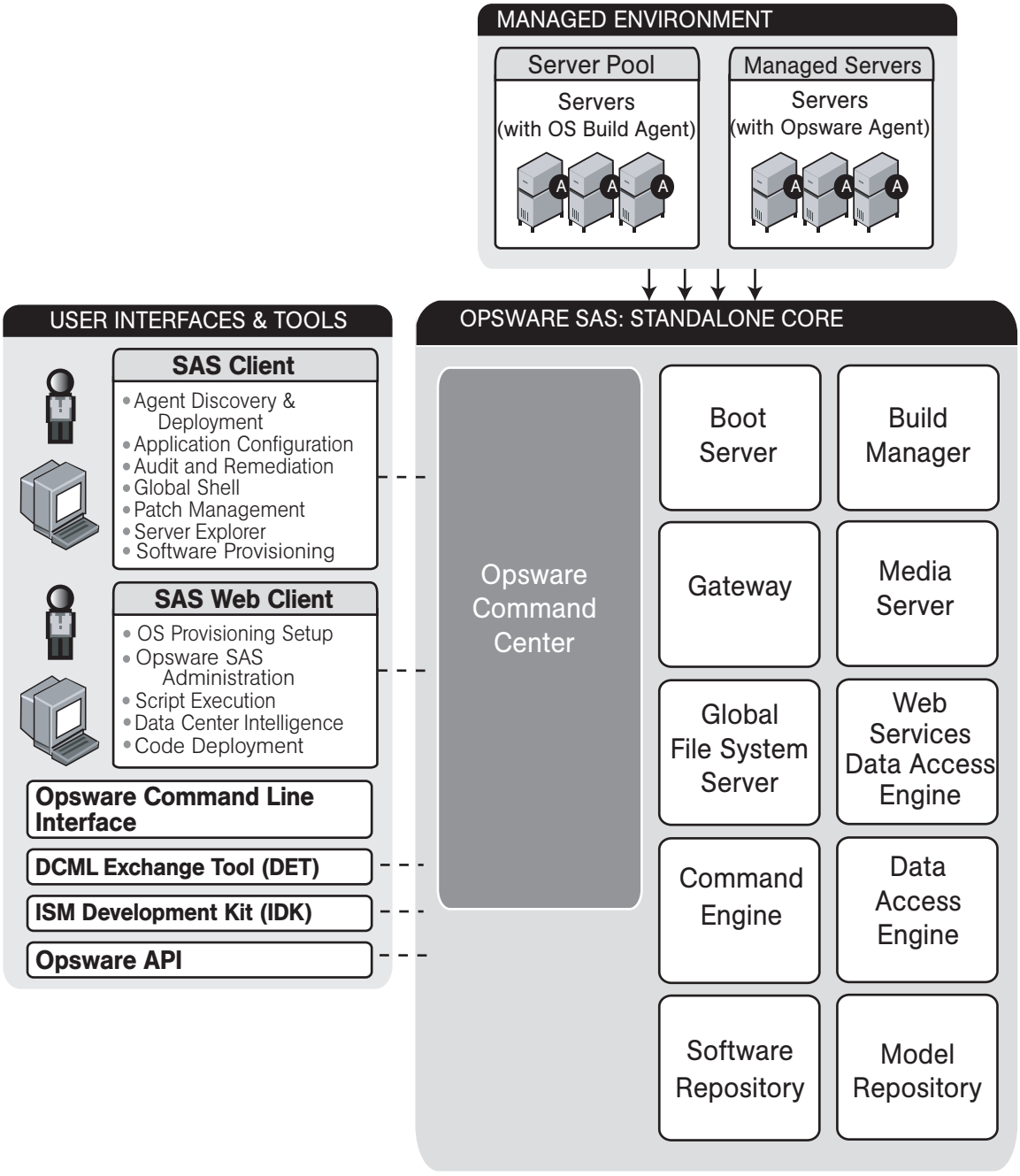
The sections that follow describe the components of Opsware SAS:

- **Boot Server**: The part of the OS Provisioning feature that supports network booting of Sun and x86 systems.
- **Build Manager**: This facilitates communication between components for OS provisioning.
- **Command Engine**: The system for running distributed programs across many servers.
- **Data Access Engine**: The XML-RPC interface to the Model Repository.
- **Media Server**: This server provides network access to vendor-supplied media used during OS provisioning.

- **Model Repository**: The Opsware SAS data repository (database).
- **Model Repository Multimaster Component**: The application that propagates and synchronizes changes from each Model Repository database to all other Model Repository databases.
- **Opsware Agents**: Intelligent agents that run on each server that Opsware SAS manages.
- **SAS Client:** The Windows user interface to Opsware SAS.
- **SAS Web Client**: The browser-based interface to Opsware SAS.
- **Opsware Command Center:** The core component that communicates with the SAS Web Client
- **OS Build Agent**: The agent responsible for registering a bare metal server with Opsware SAS and guiding the OS installation process.
- **Software Repository**: The central repository for all software that Opsware SAS manages.
- **Software Repository Replicator**: This serves as backup for Software Repositories in a multimaster mesh, ensuring that packages are available, even if one of the Software Repositories becomes unavailable.
- **Software Repository Multimaster Component**: This aids in transferring software from the Software Repository in one facility to the Software Repository in another facility in a multimaster mesh.
- **Software Repository Cache**: This contains local copies in the Opsware Satellite of the Software Repository of the core (or another Satellite).
- **Web Services Data Access Engine**: This provides increased performance from the Model Repository to other Opsware SAS components.
- **Opsware Gateway**: This provides network connectivity to Opsware cores and Satellites.
- **Global File System Server**: This hosts Global Shell sessions dynamically constructs the Opsware Global File System (OGFS), a virtual file system.

The following figure shows an overview of Opsware SAS components in a standalone core. The components in a core can be distributed across multiple servers.

Figure 1-11: Overview of the Opsware Components



Boot Server

The Boot Server, part of the OS Provisioning feature, 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, and Sun Solaris TFTP and NFS.

Build Manager

The Build Manager component facilitates communications between OS Build Agents and the Command Engine. It accepts OS provisioning commands from the Command Engine, and it provides a runtime environment for the platform-specific build scripts to perform the OS provisioning procedures.

Command Engine

The Command Engine is a system for running distributed programs across many servers (usually Opsware Agents). Command Engine scripts are written in Python and run on the Command Engine server. Command Engine scripts can issue commands to Opsware Agents. These calls are delivered in a secure manner and are auditable by using data stored in the Model Repository.

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

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.

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 Opsware SAS without requiring system-wide changes.

Media Server

The Media Server is also part of the OS Provisioning feature, and 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 NFS.

Model Repository

The Model Repository is implemented as an Oracle database. All Opsware SAS components work from, or update, a data model maintained for all servers that Opsware SAS manages. The Model Repository contains essential information necessary to build, operate, and maintain the following items:

- A list of all servers under management
- The hardware associated with these servers, including memory, CPUs, storage capacity, and so forth
- The configuration of those servers, including IP addresses
- The operating system, system software, and applications installed on servers
- Information on other software available for installation on servers and how it is bundled
- Authentication and security information

Each Opsware core, whether standalone or multimaster, contains a single Model Repository. An Opsware Satellite, which relies on a core, does not contain a Model Repository.

Model Repository Multimaster Component

The Model Repository Multimaster Component is installed in a core that belongs to a multimaster mesh. The Model Repository Multimaster Component synchronizes the data in the Model Repositories of the mesh, propagating changes from one repository to another. Every Model Repository instance has one Model Repository Multimaster Component instance. The Model Repository Multimaster Component uses TIBCO Rendezvous.

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. The receiver (Inbound Model Repository Multimaster Component) accepts the transactions and applies them to the local Model Repository.

Opsware Agents

Each server that Opsware SAS manages has an intelligent agent running on that server. The Opsware Agent is the agent of change on a server. Whenever Opsware SAS needs to make changes to servers, it does so by sending requests to the Opsware Agent.

Depending on the request, the Opsware Agent might use global Opsware SAS services (such as the Model Repository and Software Repository) in order to fulfill the request. Some functions that the Opsware Agent supports are:

- Software installation and removal
- Configuration of software and hardware
- Periodically reporting server status
- Auditing of the server

An Opsware Agent is idle unless Opsware SAS is trying to perform some change on the server. Each Opsware Agent periodically contacts the Model Repository and registers itself, which allows the Model Repository to keep track of machine status, and know when particular servers are disconnected from and reconnected to the network.

Dormant Opsware Agents

The Opsware Agent Installer can install Opsware Agents even when the Opsware SAS core is not available to a server. If a newly-installed Opsware Agent cannot contact an Opsware SAS core, the Opsware Agent runs in a dormant mode. While dormant, it periodically attempts to contact the Opsware SAS core.

When the Opsware SAS core becomes available, the Opsware Agent performs the initialization tasks, such as hardware and software registration. These usually take place when the Opsware Agent is first installed.

Opsware SAS Client

The Opsware SAS Client is a Windows user interface to Opsware SAS. With the Opsware SAS Client, the user can access most of the Opsware SAS features, including Software Management, Patch Management, Audit and Remediation, Application Configuration, and the Server Explorer.

Opsware SAS Web Client

The Opsware SAS Web Client is a browser-based user interface to Opsware SAS. Through this interface, an Opsware SAS user can use the OS Provisioning and Code Deployment and Rollback (CDR) features. An Opsware administrator manages users and defines security permissions with this interface.

Opsware Command Center

The core component that communicates with the Opsware SAS Web Client. The Opsware Command Center (OCC) includes an HTTPS proxy server and an application server. The OCC is installed on the same server as the Web Services Data Access Engine.

OS Build Agent

The OS Build Agent, part of the OS Provisioning feature, is responsible for registering bare metal servers in Opsware SAS. In addition, it is the agent of change on the server during the OS installation process until the actual Opsware Agent is installed.

Software Repository

The Software Repository is the central repository for all software that Opsware SAS manages. It contains packages for operating systems, applications (for example, BEA WebLogic or IBM WebSphere), databases, customer code, and software configuration information.

Working with the Software Repository, an Opsware Agent can install software running on the server where the Opsware Agent is installed. The Model Repository then updates its record of the software installed on the server. This process of updating the actual software configuration of a server with a specified configuration stored in the Model Repository is called remediation.

You can install new software, code, or configurations in the Software Repository by first packaging the files, and then uploading them into the Software Repository.

See the *Opsware® SAS Policy Setter's Guide* for information about how to upload software packages to the Software Repository.

Software Repository Replicator

The Software Repository Replicator provides backup functionality for Software Repositories running in a multimaster mesh. In most deployments, the Software Repositories do not all have the same content. If one of the Software Repositories becomes unavailable, this might result in some packages not being available until the Software Repository is back online.

The Software Repository Replicator provides redundant storage of Software Repositories and thereby helps to ensure that all packages remain available even when a Software Repository goes offline.

Software Repository Cache

Installed in an Opsware Satellite, a Software Repository Cache contains local copies of the contents of the Software Repository of the core (or of another Satellite). These local copies improve performance and decrease network traffic when the core installs or updates software on the managed servers in the Satellite.

Software Repository Multimaster Component

The Software Repository Multimaster Component allows software to be distributed across several Software Repositories and to be transferred from one repository to another on-demand. For example, a Solaris package that resides on Software Repository (A) is needed for installation in a second facility that contains Software Repository (B), which is part of the same multimaster mesh. The Multimaster Component allows B to discover the presence of the package on A. The package is then transferred and cached at B so that it can be used in the second facility.

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 Opsware SAS components. This object abstraction can be accessed through a Simple Object Access Protocol (SOAP) API, by third-party integration components, or by a binary protocol of Opsware SAS components like the SAS Web Client.

Opsware Gateway

The Opsware Gateway allows an Opsware core to manage servers that are behind one or more NAT devices or firewalls. Connectivity between gateways is maintained by routing messages over persistent TCP tunnels between the gateway instances.

Additionally, the gateway provides network bandwidth management between Opsware cores in a multimaster mesh and between cores and Satellites. The ability to manage network bandwidth is important when a tunnel between gateway instances transits a low-bandwidth link, which might be shared with a bandwidth-sensitive application.

One or more Opsware Gateways serve managed servers contained within an Opsware realm. In Opsware SAS, a realm is a routable IP address space, which is serviced by one or more gateways. All managed servers that connect to an Opsware core via a gateway are identified as being in that gateway's realm.

Global File System Server

The Opsware Global Shell feature runs on the Global File System Server, a core component that dynamically constructs a virtual file system – the Opsware Global File System (OGFS). The Global File System Server can connect to an Opsware Agent to open a Unix shell or to open a Windows Remote Desktop connection on a managed server.

Chapter 2: Operating System and Hardware Requirements

IN THIS CHAPTER

This section discusses the following topics:

- Supported Operating Systems: Opsware Agents and the SAS Client
- Supported Operating Systems: Opsware Core Server
- Disk Space Requirements
- Opsware Core Performance Scalability

This chapter describes the supported operating systems for Opsware SAS core servers, managed servers, and the SAS Client. This chapter also describes the hardware requirements for the servers running an Opsware SAS core and provides guidelines on how to distribute Opsware SAS components across the servers running an Opsware SAS core.

Supported Operating Systems: Opsware Agents and the SAS Client

This section lists the supported operating systems for Opsware Agents and the SAS Client.

The following table lists the supported operating systems for Opsware Agents, which run on the servers managed by Opsware SAS.

Table 2-1: Opsware Agent Supported Operating Systems

SUPPORTED OPERATING SYSTEMS FOR OPSWARE AGENT	VERSIONS	ARCHITECTURE
AIX	AIX 4.3 AIX 5.1 AIX 5.2 AIX 5.3	POWER POWER POWER POWER
HP-UX	HP-UX 10.20 HP-UX 11.00 HP-UX 11.11 HP-UX 11.23 (11i v2)	PA-RISC PA-RISC PA-RISC PA-RISC and Itanium
Sun Solaris	Solaris 6 Solaris 7 Solaris 8 Solaris 9 Solaris 10 (Update 1, Update 2, Update 3)	Sun SPARC Sun SPARC Sun SPARC Sun SPARC Sun SPARC, 64 bit x86, 32 bit x86 and Niagara
Fujitsu Solaris	Solaris 8 Solaris 9 Solaris 10	Fujitsu SPARC Fujitsu SPARC Fujitsu SPARC
Windows	Windows NT 4.0 Windows 2000 Server Family Windows Server 2003 Windows XP Professional	32 bit x86 32 bit x86 32 bit x86 and 64 bit x86 32 bit x86

Table 2-1: Opsware Agent Supported Operating Systems (continued)

SUPPORTED OPERATING SYSTEMS FOR OPSWARE AGENT	VERSIONS	ARCHITECTURE
Red Hat Linux	Red Hat Linux 7.3	32 bit x86
	Red Hat Linux 8.0	32 bit x86
	Red Hat Enterprise Linux 2.1 AS	32 bit x86
	Red Hat Enterprise Linux 2.1 ES	32 bit x86
	Red Hat Enterprise Linux 2.1 WS	32 bit x86
	Red Hat Enterprise Linux 3 AS	32 bit x86 and 64 bit x86 and Itanium
	Red Hat Enterprise Linux 3 ES	32 bit x86 and 64 bit x86 and Itanium
	Red Hat Enterprise Linux 3 WS	32 bit x86 and 64 bit x86 and Itanium
	Red Hat Enterprise Linux 4 AS	32 bit x86 and 64 bit x86
	Red Hat Enterprise Linux 4 ES	32 bit x86 and 64 bit x86
	Red Hat Enterprise Linux 4WS	32 bit x86 and 64 bit x86
	Red Hat Enterprise Linux Server 5	32 bit x86 and 64 bit x86
	Red Hat Enterprise Linux Desktop 5	32 bit x86 and 64 bit x86
SUSE Linux	SUSE Linux Enterprise Server 8	32 bit x86
	SUSE Linux Standard Server 8	32 bit x86
	SUSE Linux Enterprise Server 9	32 bit x86 and 64 bit x86
	SUSE Linux Enterprise Server 10	32 bit x86 and 64 bit x86
VMware	ESX Server 3	32 bit x86 and 64 bit x86

On Red Hat Enterprise Linux 5, Opsware does not support SELinux (Security Enhanced Linux). By default, SELinux is enabled on Red Hat Enterprise Linux 5. You must disable the SELinux feature on Red Hat Enterprise Linux 5 for the Opsware Agent to function correctly.

The following table lists the operating systems supported for the SAS Client.

Table 2-2: SAS Client Supported Operating Systems

SUPPORTED OPERATING SYSTEMS FOR SAS CLIENT	VERSIONS	ARCHITECTURE
Windows	Windows Vista Windows XP Windows 2003 Windows 2000	32 bit x86 and 64 bit x86 32 bit x86 32 bit x86 32 bit x86

For optimal performance, Opsware, Inc. recommends a minimum 1GB RAM on the system that runs the SAS Client.

Agent Installation on Windows 2000 and Windows 2003 Servers

Installation of an Opsware SAS Agent on a managed server requires the Windows Update service to be installed.

- If the service is installed, but has been disabled by the customer, the Agent will automatically start the service.
- If the service is not installed, the Agent will copy the Windows Update Agent installer to the managed server and then run it. This process will install the service and set it to automatically start on all deployed servers.

For information about the installer files for Patch Management, see “Windows Patch Management Requirements” on page 73.

If the Windows Update service is prevented from running when the agent triggers the service to start (such as, when the service is blocked by a domain policy), the following error will be reported in the managed server system log:

```
DCOM got error "The service cannot be started, either because it
is disabled or because it has no enabled devices associated with
it. " attempting to start the service wuauserv with arguments ""
in order to run the server:
{E60687F7-01A1-40AA-86AC-DB1CBF673334}
```

For more information about this error, see http://go.microsoft.com/fwlink/events.asp.

Supported Operating Systems: Opsware Core Server

This section lists the supported operating systems for Opsware core components.

For a list of supported Oracle versions for the Model Repository, see Appendix A in the *Opsware® SAS Planning and Installation Guide*.

The following table lists the supported operating systems for the Opsware core components.

Table 2-3: Opsware Core Supported Operating Systems

SUPPORTED OS FOR OPSWARE CORE	VERSIONS	ARCHITECTURE	OPSWARE COMPONENTS
Sun Solaris	Solaris 8	Sun SPARC	All components, *excluding* the Opsware Global File System Server (OGFS) component
Sun Solaris	Solaris 9	Sun SPARC	All components
Sun Solaris	Solaris 10	Sun SPARC, Niagara	All components
Red Hat Linux	Red Hat Enterprise Linux 4 AS	64 bit x86	All components
Red Hat Linux	Red Hat Enterprise Linux 3 AS	32 bit x86	All components

The following table lists the supported operating systems for the following components of an Opsware Satellite:

- Gateway
- Software Repository Cache
- Boot Server (optional)

- Media Server (optional)

Table 2-4: Opsware Satellite Supported Operating Systems

SUPPORTED OS FOR OPSWARE SATELLITE	VERSIONS	ARCHITECTURE
Sun Solaris	Solaris 9	Sun SPARC
Sun Solaris	Solaris 10	Sun SPARC
Red Hat Linux	Red Hat Enterprise Linux 3 AS	32 bit x86
Red Hat Linux	Red Hat Enterprise Linux 4 AS	64 bit x86
SUSE Linux	SUSE Linux Enterprise Server 9	32 bit x86

Disk Space Requirements

An Opsware core server is a computer running one or more Opsware core components. You can install all of the Opsware core components on a single server or you can distribute the components across multiple servers. This section describes the hardware requirements for Opsware core servers.

Core Server Disk Space Requirements

On each core server, the root directory must have at least 72 GB of hard disk space. Opsware components are installed in the `/opt/opsware` directory. Table 2-5 lists the recommended disk space requirements for installing and running Opsware components. These sizes are recommended for the primary production data. Additional storage for backups should be calculated separately.

Table 2-5: Opsware Disk Space Requirements

OPSWARE COMPONENT DIRECTORY	RECOMMENDED DISK SPACE	REQUIREMENT ORIGIN
/etc/opt/opsware	50 MB	Configuration information for all Opsware core services. (Fixed disk usage)

Table 2-5: Opsware Disk Space Requirements (continued)

OPSWARE COMPONENT DIRECTORY	RECOMMENDED DISK SPACE	REQUIREMENT ORIGIN
/media*	15 GB	The media directory holds the OS installation media that is shared over NFS or CIFS. The initial size for this directory depends on the total size of all OS installation media sets that you plan on provisioning, such as Windows 2003 CD (700mb), Redhat AS3 CDs (2GB), and Suse 9 SP3 (10GB). The network OS install shares do not need to reside on Opsware core systems and are typically dispersed across multiple servers as the Opsware mesh grows. (Bounded disk usage that grows quickly in large increments)
/opt/opsware	15 GB	The base directory for all Opsware core services. (Fixed disk usage)
/u01/oradata*	20 GB	The Oracle tablespace directory that contains all model and job history information. Known sizes range from 5GB to 50GB of space, depending on the frequency and type of work, the amount of software and servers managed, and the garbage collection frequency settings. (Bounded disk usage that grows slowly in small increments)
/var/log/opsware	10 GB	The total log space used by all Opsware components. (Fixed disk usage)
/var/opt/opsware	10 GB	The total run space used by all Opsware components, including instances, pid files, lock files, and so on. (Fixed disk usage)

Table 2-5: Opsware Disk Space Requirements (continued)

OPSWARE COMPONENT DIRECTORY	RECOMMENDED DISK SPACE	REQUIREMENT ORIGIN
/var/opt/opsware/word*	80 GB	The total disk space used by software that is imported into Opsware. Theoretically, this is infinite disk usage depending on how much software you import. Initial size calculation is based on the total size of all packages and patches that you want managed by Opsware. Known sizes range from 10GB to 250GB.
/var/opt/opsware/ogfs/mnt	20 GB	The home directory for Global Shell enabled Opsware user accounts.

* These are directory path defaults that you can change during the installation process. The recommended disk space for these directories is based on average-sized directories, which could be smaller or larger, according to usage.

The disk space requirements in Table 2-5 exclude requirements for the following components:

Model Repository (Database) Disk Space Requirements

Additional disk space is required for the Oracle software and the data files containing the Model Repository. Storage requirements for the database grow as the number of managed servers grows. As a benchmark figure, you should allow an additional 3.1 GB database storage for every 1,000 servers in the facility that Opsware SAS manages. When sizing the tablespaces, follow the general guidelines described in Table 2-6. If you need to determine a more precise tablespace sizing, contact Opsware Support.

Table 2-6: Tablespace Sizes

TABLESPACE	MB/1000 SERVERS	MINIMUM SIZE
AAA_DATA	256 MB	256 MB
AAA_INDX	256 MB	256 MB
AUDIT_DATA	256 MB	256 MB

Table 2-6: Tablespace Sizes (continued)

TABLESPACE	MB/1000 SERVERS	MINIMUM SIZE
AUDIT_INDX	256 MB	256 MB
LCREP_DATA	3,000 MB	1,500 MB
LCREP_INDX	1,600 MB	800 MB
TRUTH_DATA	1,300 MB	700 MB
TRUTH_INDX	400 MB	400 MB

Software Repository Disk Space Requirements

The Software Repository contains software packages and other installable files. Typical installations start with approximately 300 GB. However, more space might be required, depending on the number and size of the packages, as well as the frequency and duration of configuration backups.

Media Server Disk Space Requirements

This component requires sufficient disk space for the OS media it contains.

Install the Opsware components on a local disk, not on a NetApp file server. However, for the Software Repository, you can use a variety of storage solutions, including internal storage, Network Attached Storage (NAS), and Storage Area Networks (SANs).

Opsware Core Performance Scalability

You can scale the Opsware SAS core components vertically, by adding additional CPUs and memory, or horizontally, by distributing the components on multiple servers.

CPU Requirements

The CPU for core servers has the following requirements:

- Single-server: 4 dual-core CPUs (or equivalent)
- Multiple-server: 2 dual-core CPUs (or equivalent)

Memory Requirements

The memory for core servers has the following requirements:

- Single-server: 8 GB RAM (1 GB per CPU core)
- Multiple-server: 4 GB RAM (1 GB per CPU core)

Table 2-7 and Table 2-8 list the recommended distribution of Opsware components across multiple servers. The following abbreviations are used to represent Opsware components:

- CE: Command Engine
- DAE: Data Access Engine
- GW: Gateway
- OCC: Opsware Command Center
- OGFS: Opsware Global File System
- OS PBM: OS Provisioning Build Manager
- MR: Model Repository
- MR MM: Model Repository Multimaster Component
- SR: Software Repository

Table 2-7: Distribution of Core Components

NUMBER OF CORE SERVERS	OPSWARE SAS CORE COMPONENTS					
	Number of CPU Cores per Server					
	8 CPU cores	4 CPU cores	4 CPU cores	4 CPU cores	4 CPU cores	No. of Managed Servers
1	MR CE GW DAE SR OCC OSPBM OGFS MRMM					960

Table 2-7: Distribution of Core Components (continued)

NUMBER OF CORE SERVERS	OPSWARE SAS CORE COMPONENTS					
2	MR CE	GW DAE SR OCC OSPBM OGFS MRMM				2250
3	MR	GW DAE SR OCC MRMM	CE OSPBM OGFS			4500
4	MR	GW DAE SR MRMM	CE OSPBM OGFS	OCC OGFS		7200
5	MR	GW DAE SR MRMM	CE OSPBM	OCC OGFS	OCC OGFS	8000

Table 2-8: Distribution of Core Components

NUMBER OF CORE SERVERS	OPSWARE SAS CORE COMPONENTS					
	Number of CPU Cores per Server					
	4 CPU cores	4 CPU cores	4 CPU cores	4 CPU cores	4 CPU cores	No. of Managed Servers
1	MR CE GW DAE SR OCC OSPBM OGFS MRMM					480
2	MR CE	GW DAE SR OCC OSPBM OGFS MRMM				1125
3	MR	GW DAE SR OCC MRMM	CE OSPBM OGFS			2250
4	MR	GW DAE SR MRMM	CE OSPBM OGFS	OCC OGFS		3600

Table 2-8: Distribution of Core Components (continued)

NUMBER OF CORE SERVERS	OPSWARE SAS CORE COMPONENTS					
5	MR	GW DAE SR MRMM	CE OSPBM	OCC OGFS	OCC OGFS	4000

Small Core Servers

- 1 core server with 4 CPU cores, 4GB RAM: 480 managed servers
- 1 core server with 2 CPU cores, 4 GB RAM: 150 managed servers

Factors Affecting Core Performance

The hardware requirements for Opsware SAS vary based on these factors:

- The number of servers that Opsware SAS manages
- The number and complexity of concurrent operations
- The number of concurrent users accessing the Opsware Command Center
- The number of facilities in which Opsware SAS operates

Table 2-9 lists the approximate number of Opsware SAS servers per core and the CPU cores required for a given number of managed servers and Opsware users.

Table 2-9: Required Number of Core Servers

NUMBER OF MANAGED SERVERS	NUMBER OF OPSWARE USERS	NUMBER OF SAS SERVERS PER CORE & (CPU CORES)
960	40	1 (8 CPU cores)
2250	90	2 (12 CPU cores)
4500	180	3 (16 CPU cores)
7200	280	4 (20 CPU cores)
8000	300	5 (24 CPU cores)

Scalability in a Multimaster Mesh

To support global scalability, you can install an Opsware core in each major facility, linking the cores in a multimaster mesh. The size of the Opsware core in each facility can be scaled according to local requirements.

To support availability in a multimaster mesh, you can manage the servers in all facilities from a single location with the SAS Web Client or a SAS Client. Therefore, the number and location of SAS Web Client instances and SAS Clients are flexible. A common implementation is with two geographically distributed Opsware cores.

In addition to Model Repository replication, a multimaster mesh supports the replication and caching of the packages stored in the Software Repository. Typically, the Opsware core in each facility owns the software that is uploaded to the core's Software Repository. To support availability, multiple copies of the packages can be maintained in remote Software Repositories. See the *Opsware® SAS Administration Guide* for more information.

Factors Affecting Satellite Performance

Install Opsware Satellites on servers that meet the following requirements:

- 2 CPUs per 1,500 managed servers per Satellite
- 2 GB RAM per 1,500 managed servers per Satellite

Load Balancing Additional Instances of Opsware Components

If Opsware SAS needs to support a larger operational environment, you might improve performance by installing additional instances of the following core components:

- Data Access Engine
- OS Provisioning Media Server
- Opsware Command Center
- Opsware Global Filesystem

Opsware SAS does not support installing additional instances of the other components, such as the Command Engine or OS Provisioning Boot Server.

You can deploy a hardware load balancer for the servers that run additional instances of the Data Access Engine and Opsware Command Center. You can also configure the load balancer for SSL session persistence (stickiness) with the least connections algorithm. See the *Opsware® SAS Administration Guide* for the steps to install an additional instance of an Opsware SAS component.

Chapter 3: Pre-Installation Requirements

IN THIS CHAPTER

This section discusses the following topics:

- Dual Layer DVD Requirements
- Solaris and Linux Requirements
- Requirements for Installing Oracle 10g using the Opsware Installer
- Network Requirements
- Windows Patch Management Requirements
- Configuration Tracking Requirements
- Opsware Global File System (OGFS) Server Requirements
- Time and Locale Requirements

This chapter describes the system and network administration tasks that must be performed before you can run the Opsware Installer. It provides a detailed list of system requirements for Opsware core, multimaster and satellite installations, reviewing operating systems, network configuration, patch management, configuration tracking, OGFS, and time and locale requirements.

Dual Layer DVD Requirements

The Product Software DVD and the Agent and Utilities DVD require a DVD drive that supports dual layer. See “Installation Media for the Opsware Installer” on page 116 for information about the Opsware DVD set.

Solaris and Linux Requirements

This section describes platform-specific requirements. For more information about the supported operating systems for Opsware core components, see Chapter 2, “Operating System and Hardware Requirements.”

The server on which you install the Oracle RDBMS software needed by Opsware SAS has *additional* requirements, as described in “Required Operating System Packages and Patches” on page 220.

Solaris Requirements

For Solaris, the Opsware core servers must meet the requirements listed in Table 3-1, Table 3-2, and Table 3-3.

Table 3-1: Packages Required for Solaris

REQUIRED PACKAGES FOR SOLARIS		
SUNWCreq (cluster)	SUNWeurf	SUNWeudiv
SUNWadmap	SUNWi2rf	SUNWeudlg
SUNWadmc	SUNWi4rf	SUNWeudmg
SUNWdoc	SUNWi5rf	SUNWeuezt
SUNWesu	SUNWi7rf	SUNWeuhed
SUNWman	SUNWi8rf	SUNWeuluf
SUNWmkcdS	SUNWi9rf	SUNWeulux
SUNWswmt	SUNWi13rf	SUNWeuodf
SUNWtoo	SUNWi15rf	SUNWeuxwe
SUNWtoox**	SUNWtxfnt	SUNWuiu8
SUNWadmfw	SUNWinttf	SUNWuiu8x
SUNWlibC	SUNW5xmft	SUNWulcf
SUNWlibCx**	SUNWcxmft	SUNWulcfx
SUNWinst	SUNWjxmft	SUNWulocf
SUNWucbt	SUNWkxmft	SUNWuxlcf
SUNWucbtx**	SUNWeu8df	SUNWuxlcx
SUNWscpu	SUNWeu8os	SUNWeudbd
SUNWscpux**	SUNWeu8ox	SUNWeudhs
SUNWtcsh	SUNWeudba	SUNWeusru
SUNWsacom	SUNWeudda	SUNWuium
SUNWntpr	SUNWeudhr	NSCPeu8cm
SUNWntpu	SUNWeudis	
SUNWarrf		

** These packages are required only for Solaris 8 and Solaris 9.

Table 3-2: Packages Recommended for Solaris

RECOMMENDED PACKAGES FOR SOLARIS		
SUNWisolc	SUNWjiu8	SUNWkiu8x
SUNWisolx	SUNWkiu8	SUNWtiu8x
SUNWislcc	SUNWtiu8	SUNWi1of
SUNWislcx	SUNWciu8x	SUNWiniu8
SUNWciu8	SUNWhiu8x	SUNWiniu8x
SUNWhiu8	SUNWjiu8x	

Table 3-3: Packages That Must Be Removed from Solaris

PACKAGES THAT MUST BE REMOVED FROM SOLARIS
SUNWCpm

Other Solaris Requirements

For Solaris, the Opsware core servers must also meet the following requirements:

- On the server where you will install the SAS Web Client component, you must install the J2SE Cluster Patches for Solaris. To download these patches, search for "J2SE Cluster Patches" for your version of Solaris at `http://www.sun.com/`.
- On all core servers, verify that the Network File System (NFS) is configured and running.
- For Daylight Saving Time (DST) on Solaris 9 servers, you must install the time zone patch 113225-07 or later, and libc patch 112874-33 or later. To download these patches, search for the patch ID at `http://www.sun.com/`.
- For Daylight Saving Time (DST) on Solaris 10 servers, you must install the time zone patch 122032-03 or later, and libc patch 119689-07 or later. To download these patches, search for the patch ID at `http://www.sun.com/`.

  For more information about DST changes, search for "Daylight Saving Time (DST)" at `http://www.sun.com/`.

If you attempt to download any of these files and receive an error page indicating that the file was not found, make sure you are using the correct URL. For the correct URL, check the Opsware Technical Support web site at https://download.opsware.com. For instructions, contact support@opsware.com.

Linux Requirements

For Linux AS3 32-bit x86 and Linux AS4 64-bit x86, the Opsware core servers must meet the requirements listed in Table 3-4, Table 3-5, and Table 3-6.

Due to a known Linux AS4 64-bit x86 kernel bug, you must have Update 5 or later installed on all Opsware core servers.

Table 3-4: Packages Required for Linux AS3 32-bit x86

REQUIRED PACKAGES FOR LINUX AS3 32-BIT X86		
at	iptables	patch
compat-db	kernel-source	patchutils
compat-libstdc++	libcap	sharutils
coreutils	libxml2-python	strace
cpp	libstdc++	tcl
expat	libstdc++-devel	unzip
gcc	mkisofs	XFree86-libs
glibc-devel	ncompress (contains	XFree86-libs-data
glibc-headers	uncompress utility)	XFree86-Mesa-libGL
glibc-kernheaders	nfs-utils	xinetd
	ntp	zip

Table 3-5: Packages Required for Linux AS4 64-bit x86

REQUIRED PACKAGES FOR LINUX AS4 64-BIT X86
binutils-2.15.92.0.2-21.x86_64.rpm
chkfontpath-1.10.0-2.x86_64.rpm
compat-db-4.1.25-9.i386.rpm
compat-db-4.1.25-9.x86_64.rpm
compat-libstdc++-33-3.2.3-47.3.i386.rpm
compat-libstdc++-33-3.2.3-47.3.x86_64.rpm
control-center-2.8.0-12.rhel4.5.x86_64.rpm
cpp-3.4.6-3.x86_64.rpm
desktop-file-utils-0.9-2.x86_64.rpm
expat-1.95.7-4.i386.rpm
expat-1.95.7-4.x86_64.rpm
expat-devel-1.95.7-4.x86_64.rpm
gamin-0.1.1-4.EL4.i386.rpm
gamin-0.1.1-4.EL4.x86_64.rpm
gamin-devel-0.1.1-4.EL4.x86_64.rpm
gamin-python-0.1.1-4.EL4.x86_64.rpm
gamin-devel-0.1.1-4.EL4.x86_64.rpm
gcc-3.4.6-3.x86_64.rpm
gcc-c++-3.4.6-3.x86_64.rpm
gcc-g77-3.4.6-3.x86_64.rpm
gcc-gnat-3.4.6-3.x86_64.rpm
gcc-java-3.4.6-3.x86_64.rpm
gcc-objc-3.4.6-3.x86_64.rpm
gcc4-4.1.0-18.EL4.x86_64.rpm
gcc4-c++-4.1.0-18.EL4.x86_64.rpm
gcc4-gfortran-4.1.0-18.EL4.x86_64.rpm
gcc4-java-4.1.0-18.EL4.x86_64.rpm
gcc-c++-3.4.6-3.x86_64.rpm
glibc-2.3.4-2.25.i686.rpm
glibc-2.3.4-2.25.x86_64.rpm
glibc-common-2.3.4-2.25.x86_64.rpm
glibc-devel-2.3.4-2.25.i386.rpm
glibc-devel-2.3.4-2.25.x86_64.rpm
glibc-headers-2.3.4-2.25.x86_64.rpm
glibc-kernheaders-2.4-9.1.98.EL.x86_64.rpm

Table 3-5: Packages Required for Linux AS4 64-bit x86 (continued)

REQUIRED PACKAGES FOR LINUX AS4 64-BIT X86
glibc-profile-2.3.4-2.25.x86_64.rpm
glibc-utils-2.3.4-2.25.x86_64.rpm
glibc-common-2.3.4-2.25.x86_64.rpm
glibc-devel-2.3.4-2.25.i386.rpm
glibc-devel-2.3.4-2.25.x86_64.rpm
glibc-headers-2.3.4-2.25.x86_64.rpm
glibc-kernheaders-2.4-9.1.98.EL.x86_64.rpm
gnome-libs-1.4.1.2.90-44.1.x86_64.rpm
gnome-libs-devel-1.4.1.2.90-44.1.x86_64.rpm
kernel-smp-2.6.9-55.EL.x86_64.rpm
kernel-smp-devel-2.6.9-55.EL.x86_64.rpm
libaio-0.3.105-2.i386.rpm
libaio-0.3.105-2.x86_64.rpm
libaio-devel-0.3.105-2.x86_64.rpm
libcap-1.10-20.i386.rpm
libcap-1.10-20.x86_64.rpm
libcap-devel-1.10-20.x86_64.rpm
libgcc-3.4.6-3.i386.rpm
libgcc-3.4.6-3.x86_64.rpm
libpng-1.2.7-1.el4.2.i386.rpm
libpng-1.2.7-1.el4.2.x86_64.rpm
libpng-devel-1.2.7-1.el4.2.x86_64.rpm
libpng10-1.0.16-1.i386.rpm
libpng10-1.0.16-1.x86_64.rpm
libpng10-devel-1.0.16-1.x86_64.rpm
libstdc++-3.4.6-3.i386.rpm
libstdc++-3.4.6-3.x86_64.rpm
libstdc++-devel-3.4.6-3.i386.rpm
libstdc++-devel-3.4.6-3.x86_64.rpm
libstdc++-devel-3.4.6-3.i386.rpm
libstdc++-devel-3.4.6-3.x86_64.rpm
libtermcap-2.0.8-39.i386.rpm
libtermcap-2.0.8-39.x86_64.rpm

Table 3-5: Packages Required for Linux AS4 64-bit x86 (continued)

REQUIRED PACKAGES FOR LINUX AS4 64-BIT X86
libtermcap-devel-2.0.8-39.x86_64.rpm
libxml2-2.6.16-6.i386.rpm
libxml2-2.6.16-6.x86_64.rpm
libxml2-devel-2.6.16-6.x86_64.rpm
libxml2-python-2.6.16-6.x86_64.rpm
make-3.80-6.EL4.x86_64.rpm
mkisofs-2.01.1-5.x86_64.rpm
ncompress-4.2.4-41.rhel4.x86_64.rpm
nfs-utils-1.0.6-70.EL4.x86_64.rpm
nfs-utils-lib-1.0.6-3.x86_64.rpm
nfs-utils-lib-devel-1.0.6-3.x86_64.rpm
ntp-4.2.0.a.20040617-4.EL4.1.x86_64.rpm
openmotif21-2.1.30-11.RHEL4.6.i386.rpm
patch-2.5.4-20.x86_64.rpm
patchutils-0.2.30-1.x86_64.rpm
pdksh-5.2.14-30.3.x86_64.rpm
popt-1.9.1-18_nonptl.i386.rpm
popt-1.9.1-18_nonptl.x86_64.rpm
readline-4.3-13.i386.rpm
readline-4.3-13.x86_64.rpm
readline-devel-4.3-13.x86_64.rpm
rpm-build-4.3.3-18_nonptl.x86_64.rpm
screen-4.0.2-5.x86_64.rpm
sharutils-4.2.1-22.2.x86_64.rpm
strace-4.5.14-0.EL4.1.x86_64.rpm
switchdesk-4.0.6-3.noarch.rpm
switchdesk-gui-4.0.6-3.noarch.rpm
sysstat-5.0.5-11.rhel4.x86_64.rpm
tcl-8.4.7-2.i386.rpm
tcl-8.4.7-2.x86_64.rpm
tcl-devel-8.4.7-2.x86_64.rpm
tcl-html-8.4.7-2.x86_64.rpm
tclx-8.3.5-4.i386.rpm
tclx-8.3.5-4.x86_64.rpm
tclx-devel-8.3.5-4.x86_64.rpm

Table 3-5: Packages Required for Linux AS4 64-bit x86 (continued)

REQUIRED PACKAGES FOR LINUX AS4 64-BIT X86
tclx-doc-8.3.5-4.x86_64.rpm
tcp_wrappers-7.6-37.2.i386.rpm
tcp_wrappers-7.6-37.2.x86_64.rpm
ttmkfdir-3.0.9-14.1.EL.x86_64.rpm
unzip-5.51-7.x86_64.rpm
vim-enhanced-6.3.046-0.40E.7.x86_64.rpm
vnc-4.0-8.1.x86_64.rpm
vnc-server-4.0-8.1.x86_64.rpm
xinetd-2.3.13-4.4E.1.x86_64.rpm
xinitrc-4.0.14.3-1.noarch.rpm
xorg-x11-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-Mesa-libGL-6.8.2-1.EL.13.36.i386.rpm
xorg-x11-Mesa-libGL-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-Mesa-libGLU-6.8.2-1.EL.13.36.i386.rpm
xorg-x11-Mesa-libGLU-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-Xdmx-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-Xnest-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-Xvfb-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-deprecated-libs-6.8.2-1.EL.13.36.i386.rpm
xorg-x11-deprecated-libs-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-deprecated-libs-devel-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-devel-6.8.2-1.EL.13.36.i386.rpm
xorg-x11-devel-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-doc-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-font-utils-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-libs-6.8.2-1.EL.13.36.i386.rpm
xorg-x11-libs-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-sdk-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-tools-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-twm-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-xauth-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-xdm-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-xfs-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-Mesa-libGL-6.8.2-1.EL.13.36.i386.rpm

Table 3-5: Packages Required for Linux AS4 64-bit x86 (continued)

REQUIRED PACKAGES FOR LINUX AS4 64-BIT X86
xorg-x11-Mesa-libGL-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-Mesa-libGLU-6.8.2-1.EL.13.36.i386.rpm
xorg-x11-Mesa-libGLU-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-Xvfb-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-deprecated-libs-6.8.2-1.EL.13.36.i386.rpm
xorg-x11-deprecated-libs-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-deprecated-libs-devel-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-font-utils-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-libs-6.8.2-1.EL.13.36.i386.rpm
xorg-x11-libs-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-twm-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-xauth-6.8.2-1.EL.13.36.x86_64.rpm
xorg-x11-xfs-6.8.2-1.EL.13.36.x86_64.rpm
xscreensaver-4.18-5.rhel4.11.x86_64.rpm
xterm-192-4.EL4.x86_64.rpm
zip-2.3-27.x86_64.rpm
zlib-1.2.1.2-1.2.i386.rpm
zlib-1.2.1.2-1.2.x86_64.rpm
zlib-devel-1.2.1.2-1.2.i386.rpm
zlib-devel-1.2.1.2-1.2.x86_64.rpm

Table 3-6: Packages That Must Be Removed for Linux

PACKAGES THAT MUST BE REMOVED FROM LINUX		
samba	rsync	tftp**
apache	httpd	dhcp**

To verify that the `samba` package, for example, is installed, enter the following command:

```
rpm -qa | grep samba
```

You can obtain the latest versions of these packages from the Red Hat errata web site.

** Existing versions of the `tftp` and `dhcp` packages cannot reside on the same server as the OS Provisioning Boot Server component; however, they can reside on Opsware core servers that do not have the OS Provisioning Boot Server component.

To remove packages, enter the following command:

```
rpm -e package_name
```

Some packages in this list may be depended on by other packages that are installed on your system. For example, the default Red Hat installation includes `mod_python` and `mod_perl` that rely on `httpd` being installed. In order to remove packages that fulfill dependencies, you must simultaneously remove the packages that create the dependencies. In this example, you would need to enter the following command:

```
rpm -e httpd mod_python mod_perl
```

If `rpm` identifies an additional dependency, it will note which packages have dependencies on the components to be removed and fail. These packages must be added to the uninstall command line. If the chain of dependencies cannot be suitably resolved, enter the `rpm -e --nodeps` command to remove the desired packages without considering dependencies.

Other Linux Requirements

For Linux systems, you must also perform the following tasks:

- Change the initial run level of the server to level 3 in the file `/etc/inittab`.
- If the server uses Integrated Drive Electronics (IDE) hard disks, enable Direct Memory Access (DMA) and some other advanced hard disk features to improve performance. Run the following script as root on the server and then reboot the server:

  ```
  cat > /etc/sysconfig/harddisks << EOF
  USE_DMA=1
  MULTIPLE_IO=16
  EIDE_32BIT=3
  LOOKAHEAD=1
  EOF
  ```

- Due to a bug in the Linux kernel, the loopback interface must be configured to use a maximum transmission unit (MTU) size of 16036 bytes or less. To make this change, perform the following steps:

  1. Run the `ifconfig lo mtu 16036` command. This sets the MTU of the running kernel.

2. Add the line `MTU=16036` to the end of the `/etc/sysconfig/network-scripts/ifcfg-lo` file. This causes the MTU to be properly set when the system is booted.

- Disable the Security-Enhanced Linux kernel (SELinux) on all core servers running Linux AS4 64-bit x86.
- For Daylight Saving Time (DST) on Red Hat Enterprise Linux AS 3, you must apply the tzdata-2006m-3... updates. You can download these updates from the following location:

  ```
  https://rhn.redhat.com/errata/RHEA-2006-0745.html
  ```

- For Daylight Saving Time (DST) on SuSE Linux Enterprise Server 9, you must apply the time zone-2.3.3-98.82... updates. You can download these updates from the following location:

  ```
  http://www.novell.com/support/
  dynamickc.do?externalId=3853518&sliceId=SAL_
  Public&command=show&forward=nonthreadedKC&kcId=3853518
  ```

- By default, Linux enables NFSv3, which prevents Solaris servers from entering the server pool. If your nfs server is a Linux machine. You can either add dhcp options to force Solaris 10 to use NFSv2 or you can disable NFSv3 on the Linux server.
  - To force the solaris miniroot to use NFSv2, add the following lines to your dhcp configuration file:
    1. In the main section of the dhcp configuration file, add the following lines:

       ```
       # added for nfs 2 miniroot
       option SUNW.SrootOpt code 1 = text;
       # end of nfs 2 miniroot stuff
       ```

    2. In the solaris-sun4u solaris-sun4us and solaris-specific-kernel classes, add the following lines:

       ```
       # added for nfs 2 miniroot
       option SUNW.SrootOpt "vers=2";
       # end of nfs 2 miniroot stuf
       ```

- Instead of adding lines to your dhcp configuration file, you can disable NFSv3 on the Linux media server by adding the following lines in `/etc/sysconfig/nfs` and then restart nfs:

  ```
  MOUNTD_NFS_V3=no
  MOUNTD_NFS_V2=yes
  ```

Requirements for Installing Oracle 10g using the Opsware Installer

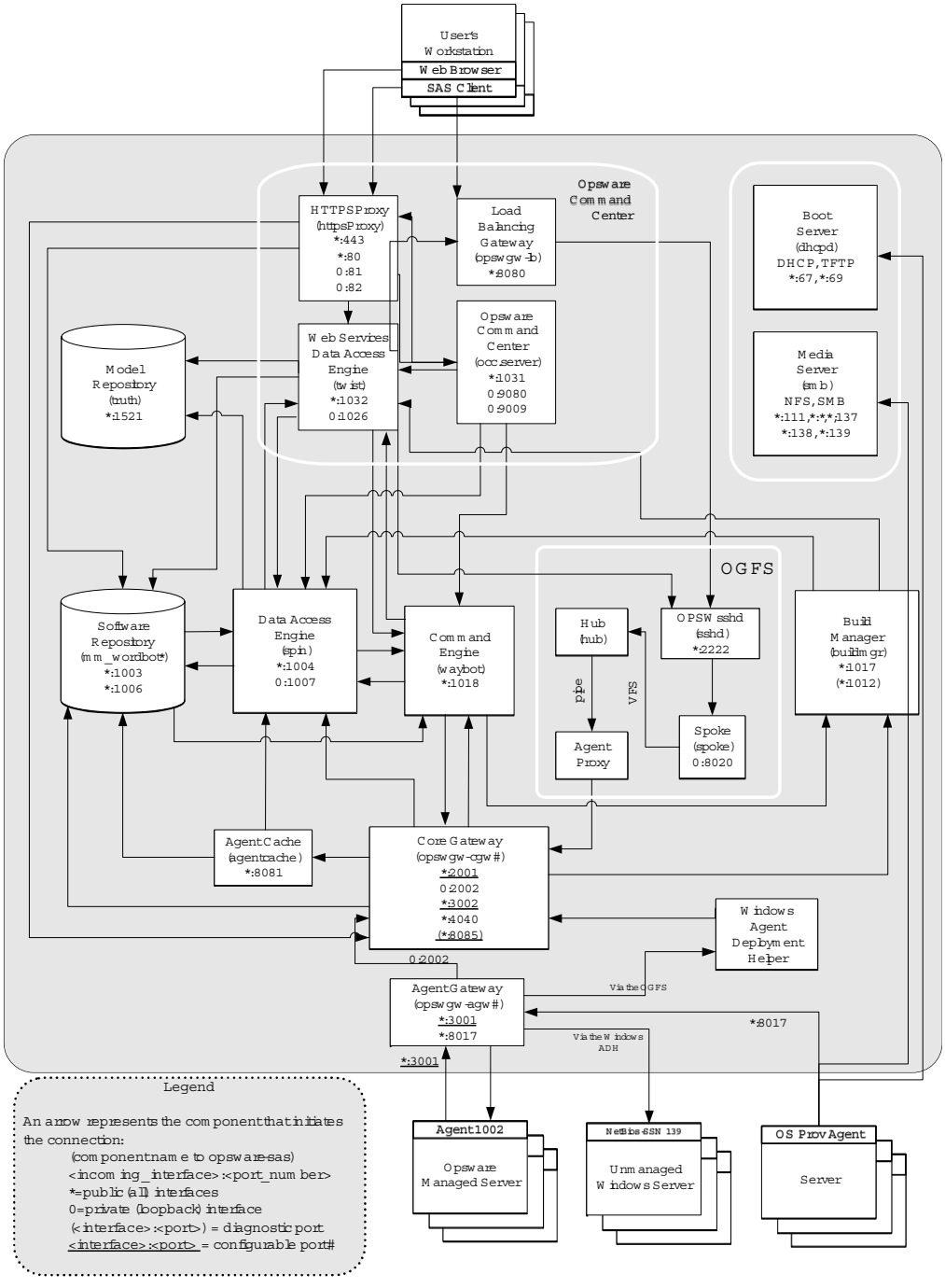
The Opsware Model Repository relies on an Oracle database. If you use the Opsware Installer to install Oracle 10g on a Solaris 8, Solaris 9, or Solaris 10 server, see "Database Parameter Value Differences" on page 224

Network Requirements

This section discusses the network requirements within a facility, open ports required for core components, and name resolution requirements. These requirements must be met for both standalone, multimaster, and Satellite cores.

Figure 3-1 shows the networking requirements for an Opsware SAS single core configuration. In a multi-server installation, there can be multiple instances of the Data Access Engine (spin) and Web Services Data Access Engine (twist) so that internal components access them over loopbacks.

Figure 3-1: Network Requirements for a Single Opsware SAS Core



Network Requirements within a Facility

Before running the Opsware Installer, your environment must meet the following network requirements:

- The Opsware core servers must be on the same Local Area Network (LAN or VLAN).
- The Opsware core servers must have network connectivity to the servers that the Opsware core manages, and vice versa.
- The Opsware core servers cannot use the Network Information Service (NIS) for password and group databases. The Opsware components check for the existence of certain target accounts before creating them during installation.
- When using network storage for Opsware components, such as the Software Repository or Media Server, the network storage configuration must allow the root user to have write access over NFS to the directories where the components are to be installed.
- The speed and duplex mode of the NIC adapters of the Opsware core and managed servers must match the switch they are connected to. A mismatch causes poor network performance between the core and managed servers, making Opsware SAS unusable.

Open Ports

Table 3-7 shows the ports that must be open on firewalls that protect the Opsware core components. The Gateway ports listed are the default values, which can be changed during the installation.

Table 3-7: Open Ports on a Firewall Protecting an Opsware Core

PORT	COMPONENT	PURPOSE
80 (TCP)	Opsware Command Center	HTTP redirector
443 (TCP)	Opsware Command Center	HTTPS Proxy for SAS Web Client UI, SAS Client, Opsware Web Services (2.2)
2001 (TCP)	Core Gateway	Inbound tunnels from other Gateways
2222 (TCP)	Opsware Global File System	Global shell session from an SSH client
3001 (TCP)	Agent Gateway	Inbound Agent connections

Table 3-7: Open Ports on a Firewall Protecting an Opsware Core (continued)

PORT	COMPONENT	PURPOSE
7580, 7581 (TCP)	Model Repository Multimaster Component	TIBCO Rendezvous web client
8017 (UDP, TCP)	Agent Gateway	Interface to the Build Manager
8080 (TCP)	Opsware Command Center	Load Balancing Gateway for the SAS Client

Table 3-8 shows the ports for the OS provisioning components that are accessed by servers during the provisioning process. (In Opsware SAS, provisioning refers to the installation of an operating system on a server.)

Table 3-8: Open Ports for the OS Provisioning Components

PORT	COMPONENT	SERVICE
67 (UDP)	Boot Server	DHCP
69 (UDP)	Boot Server	TFTP
111 (UDP, TCP)	Boot Server, Media Server	RPC (`portmapper`), required for NFS
Dynamic*	Boot Server, Media Server	`rpc.mountd`, required for NFS
2049 (UDP, TCP)	Boot Server, Media Server	NFS

* The `rpc.mountd` process runs on a dynamic port and is not fixed. Therefore, if a firewall is in place, it must be an application layer firewall that can understand the RPC request that the client uses to locate the port for `mountd`. The firewall must dynamically open that port.

Table 3-9 shows the ports that must be open on managed servers so that Opsware core servers can connect to managed servers.

Table 3-9: Open Ports on Managed Servers

PORT	COMPONENT
1002 (TCP)	Opsware Agent

Host and Service Name Resolution Requirements

Opsware SAS must be able to resolve Opsware server host names and service names to IP addresses through configuration of DNS or `/etc/hosts`.

Previous Releases

If you are installing Opsware components on servers where a previous release of Opsware SAS was installed (for example, 4.0), you must verify that the host names and service names resolve correctly as noted in this section.

Opsware Core Servers and Name Resolution

An Opsware core server must be able to resolve the fully qualified host name of itself and any other Opsware core server. (A fully qualified name includes the subdomain, for example, `myhost.acct.buzzcorp.com`.) Enter the `hostname` command and verify that it displays the fully qualified name.

Additionally, an Opsware core server must be able to resolve both the fully qualified and unqualified names of the Opsware services. (Each service name represents an Opsware component.) For example, both `truth` (unqualified) and `truth.acct.buzzcorp.com` (fully qualified) must resolve to the IP address of the server containing the Model Repository. The list of fully qualified names of the Opsware services follows:

- `truth.`*`subdomain`* – Model Repository
- `way.`*`subdomain`* – Command Engine
- `spin.`*`subdomain`* – Data Access Engine
- `theword.`*`subdomain`* – Software Repository
- `twist.`*`subdomain`* – Web Services Data Access Engine
- `occ.`*`subdomain`* – Opsware Command Center
- `buildmgr.`*`subdomain`* – OS Provisioning Build Manager
- `wordcache.`*`subdomain`* – Software Repository Multimaster Component (The name `wordcache` must resolve to the core server running the Software Repository.)

The Software Repository server must be able to resolve the IP address to the host name of the OGFS server. To enable this reverse lookup, configure DNS.

On Solaris 10, an OGFS installation requires the real host name of the OGFS server. In the `dfstab` file on the Software Repository server, specify that the first host is the real host name of the OGFS server.

DHCP Proxying

If network provisioning occurs on a separate network from the Opsware core components, you must set up DHCP proxying (for example, with Cisco IP Helper) to the DHCP server. If you set up DHCP proxying, the server/router performing the DHCP proxying must be the router for the network so that PXE will function correctly in the Opsware OS Provisioning Feature.

The Opsware Boot Server component includes a DHCP server, but does not include a DHCP proxy. You configure the DHCP server after installation by using the Opsware DHCP Network Configuration Tool. See *DHCP Configuration for OS Provisioning* in Chapter 7, on page 144.

DMZ Network

The Boot Server and Media Server run various services (such as portmapper and rpc.mountd) that have been susceptible to network attacks. Opsware Inc. recommends that you segregrate the OS Provisioning Boot Server and Media Server components onto their own DMZ network. When you segregate these components, the ports listed previously) should be opened to the DMZ network from the installation client network. Additionally, the Boot Server and Media Server should have all vendor-recommended security patches applied.

Windows Patch Management Requirements

For Windows Patch Management, you must obtain several files from Microsoft and copy them to a directory that is accessible by the Opsware Installer. When you install the Opsware Software Repository, the Opsware Installer prompts you for the directory name.

To obtain these files, perform the following steps:

1. Obtain the following files from Microsoft:

   - `qchain.exe`

     The `qchain.exe` utility is a command-line program that chains hotfixes together. To download the package containing `qchain.exe`, search for "`qchain.exe`" at `http://www.microsoft.com`. Install the package on a Windows machine and note the location of the `qchain.exe` file.

- `wsusscn2.cab`

  The `wsusscn2.cab` file contains the Microsoft patch database. To download the package containing `wsusscn2.cab`, search for "`wsusscn2.cab`" at `http://www.microsoft.com`.

- `WindowsUpdateAgent20-x86.exe`

  The `WindowsUpdateAgent20-x86.exe` file is required by the `mbsacli.exe` utility. To download the package containing `WindowsUpdateAgent20-x86.exe`, search for "`WindowsUpdateAgent20-x86.exe`" at `http://www.microsoft.com`.

- `WindowsUpdateAgent20-x64.exe`

  The `WindowsUpdateAgent20-x64.exe` file is required by the `mbsacli.exe` utility. To download the package containing `WindowsUpdateAgent20-x64.exe`, search for "`WindowsUpdateAgent20-x64.exe`" at `http://www.microsoft.com`.

- `mbsacli.exe`

  Packaged with the MBSA 1.2.1 software, the `mbsaclie.exe` utility is a command-line program that performs security scans. To download the package containing `MBSA 1.2.1`, search for "`MBSA 1.2.1`" at `http://www.microsoft.com`.

  After the download, on a Windows machine run `MBSASetup-EN.msi` to install MBSA 1.2.1.

  In the directory where you installed MBSA 1.2.1, note the location of the `mbascli.exe` file. By default, the file is installed here:

  ```
  %program files%\Microsoft Baseline Security
  Analyzer\mbascli.exe
  ```

- `mbsacli20.exe`

  This file is packaged with MBSA 2.0 that you download by searching for "`MBSA 2.0`" at `http://www.microsoft.com`.

  After the download, on a Windows machine run `MBSASetup-EN.msi` to install MBSA 2.0. In the directory where you installed MBSA 2.0, locate the `mbsacli.exe` file. By default, the file is installed here:

  ```
  %program files%\Microsoft Baseline Security Analyzer
  2\mbsacli.exe
  ```

  Copy the `mbsacli.exe` file of MBSA 2.0 to a new file named `mbsacli20.exe`.

- `wusscan.dll`

The `wusscan.dll` file is in the directory where you installed MBSA 2.0. By default, the file is here:

```
%program files%\Microsoft Baseline Security Analyzer
2\wusscan.dll
```

2. Copy the files you obtained in the preceding step. Put them in a directory that is accessible by the server where you will install the Software Repository. For example, you might copy the files to the following directory:

   `/home/win_util`

3. Verify that the destination directory contains the following files:

   ```
   mbsacli.exe
   mbsacli20.exe
   WindowsUpdateAgent20-x86.exe
   WindowsUpdateAgent20-x64.exe
   qchain.exe
   wsusscn2.cab
   wusscan.dll
   ```

4. Write down the name of the directory containing the files. When you install the Software Repository, you are prompted for the directory name. The Opsware Installer prompt is `windows_util_loc`.

During Opsware Agent installation, the files you obtained from Microsoft are downloaded from the Software Repository to the appropriate Windows servers. If newer versions of the files are uploaded to the Software Repository, they are downloaded to the managed servers during software registration. After the core is installed and running, you can upload new versions of these files with the Patch Settings window of the SAS Client. For more information, see "Agent Installation on Windows 2000 and Windows 2003 Servers" on page 44.

Configuration Tracking Requirements

When you run the Opsware Configuration Tracking feature in a facility, a separate partition is created on the server running the Software Repository for the following Configuration Tracking directory:

`/var/opt/opsware/word/<facility-name>/acsbar`

You can specify the Software Repository root directory at installation time. The default is `/var/opt/opsware/word`.

The Configuration Tracking feature uses this directory to store the backup versions of tracked configuration files and databases. The configuration backup directory is relative to the Software Repository root directory, such as:

```
<word_root>/<facility_name>/acsbar
```

Opsware Global File System (OGFS) Server Requirements

This section discusses requirements of the OGFS server.

OGFS Store and Audit Hosts

When you run the Opsware Installer interviewer in advanced mode, you can specify values for the `ogfs.store.host` and `ogfs.audit.host` parameters. (See "Opsware Global File System Prompts" on page 113.) If you set either of these parameters to a host that runs neither the OGFS nor the Software repository, then perform the following steps on the host (Solaris or Linux) where you will install the OGFS:

1. With `mkdir`, create the directories that you specified for the `ogfs.store.path` and `ogfs.audit.path` parameters.
2. Modify the export tables.

   1. On a Solaris host, modify the `/etc/dfs/dfstab` file, such as:

      ```
      # Begin Opsware ogfs export
      share -F nfs -o anon=0,rw=1.2.3.4:1.2.3.5 /export/ogfs/
      store
      share -F nfs -o anon=0,rw=1.2.3.4:1.2.3.5 /export/ogfs/
      audit
      # End Opsware ogfs exports
      ```

      where 1.2.3.4 and 1.2.3.5 are example IP addresses of the two OGFS hosts and where `/export/ogfs/store` and `/export/ogfs/audit` are corresponding paths that exist on the host from where you are exporting the OGFS data.

   2. On a Linux host, modify the `/etc/exports` file, such as:

      ```
      # Begin Opsware ogfs export
      /export/ogfs/store 1.2.3.4(rw,no_root_squash,sync) \
      1.2.3.5(rw,no_root_squash,sync)
      /export/ogfs/audit 1.2.3.4(rw,no_root_squash,sync) \
      1.2.3.5(rw,no_root_squash,sync)
      # End Opsware ogfs exports
      ```


where 1.2.3.4 and 1.2.3.5 are example IP addresses of the two OGFS hosts and where `/export/ogfs/store` and `/export/ogfs/audit` are corresponding paths that exist on the host from where you are exporting the OGFS data.

In these examples, the OGFS component is installed on two separate hosts within the same core.

3 After you add new entries to the export tables, export the directories or restart the Network File System using standard system procedures.

Remember to verify that the NFS Daemon will start when the system reboots.

Name Service Caching Daemon (nscd) and OGFS

If the Name Service Caching Daemon (`nscd`) runs on the same server as the OGFS, then users cannot open a global shell session with a direct `ssh` connection. If `ncsd` is running on the OGFS server, the Opsware Installer turns it off and runs the `chkconfig nscd off` command to prevent it from starting after a reboot. No action is required.

Time and Locale Requirements

This section discusses the time and locale requirements for core servers.

Core Time Requirements

Opsware core servers (either standalone or multimaster) and Opsware Satellite servers must meet the following requirements. These time requirements do not apply to managed servers (that is, servers with Opsware Agents).

- Opsware core servers must maintain synchronized clocks. For example, you can synchronize the system clocks with an external server that uses NTP (Network Time Protocol) services.
- Opsware core servers must have their time zone set to Coordinated Universal Time (UTC).

To configure the time zone on Linux servers, perform the following steps:

1. Copy or link `/usr/share/zoneinfo/UTC` to `/etc/localtime`.
2. Make sure that `/etc/sysconfig/clock` contains the following lines:

```
ZONE="UTC"
UTC=true
```

To configure the time zone on Solaris servers, verify that `/etc/TIMEZONE` contains the following line: TZ=UTC.

Locale Requirements

The core servers with the Model Repository and the Software Repository must have the `en_US.UTF-8` locale installed. To display data from managed servers in various locales, the core server with the Opsware Global File System (OGFS) must have those locales installed. To enable non-English locales for Windows patching, follow the instructions in "Locales for Windows Patching" in the *Opsware® SAS User's Guide: Application Automation*.

To verify whether the `en_US.UTF-8` locale is specified for core servers, enter the following command:

```
echo $LANG
```

To define or modify this locale, enter the following values in `/etc/sysconfig/i18n`:

```
LANG="en_US.UTF-8"
SUPPORTED="en_US.UTF-8:en_US:en"
```


Chapter 4: Installation Methods and Checklists

IN THIS CHAPTER

This section discusses the following topics:

- Types of Opsware SAS Installations
- Opsware Core Installation Process Flow
- Installation Checklists

The chapter reviews the types of Opsware installations, gives a general outline of the core installation process, and provides checklists that will help you prepare for and complete the installation process.

Types of Opsware SAS Installations

There are three basic types of Opsware SAS installations: standalone, multimaster, and satellite.

- **Standalone**: A standalone core does not communicate or exchange information with other cores. A standalone core manages servers in a single facility. (Optionally, a standalone core can also manage servers in remote facilities installed with Opsware Satellites.) A core contains all components of Opsware SAS, except for the Opsware Agents, which run on the servers managed by the core.
- **Multimaster**: A multimaster core exchanges information with other cores. This collection of cores is called a multimaster mesh. With a multimaster mesh, you can centralize the management of several facilities, but still get the performance benefits of having a local copy of key Opsware SAS data at each facility.
- **Satellite**: Installed in a remote facility, an Opsware Satellite provides network connection and bandwidth management for a core that manages remote servers. A Satellite must be linked to at least one core, which may be either standalone or multimaster.

This guide uses the term facility to refer to the collection of servers and devices that reside in a single physical location. A facility can be all or part of a data center, server room, or computer lab. Each Opsware core or Satellite is associated with a specific facility.

Opsware Core Installation Process Flow

There are six main phases to the Opsware core installation process. These phases are explained in general in the following steps. To get more detailed information, see the cross reference associated with each step.

1. **Planning**: In the planning phases, choose the type of Opsware SAS installation appropriate for your site and the necessary hardware. At the end of this phase, you may follow the instructions in this installation guide.

   See Chapter 1, “Opsware SAS Architecture” on page 19 of this guide for more information.

   See Chapter 2, “Operating System and Hardware Requirements” on page 41 of this guide for more information.

2. **Pre-installation Requirements**: Perform hands-on administrative tasks such as resolving host names, opening ports, and installing the necessary OS utilities or patches.

   See Chapter 3, “Pre-Installation Requirements” on page 55 of this guide for more information.

3. **Pre-requisite Information for Installer Interview**: Gather information for the Opsware Installer interview, which will prompt you for information about the core and your operational environment. This information includes the name of the facility to be managed by the core, the authorization domain, and information about the Oracle database that underlies the Opsware Model Repository.

   See Chapter 5, “Prerequisites for the Installer Interview” on page 91 of this guide for more information.

4. **Perform Installation**: Run the Opsware Installer, complete the interview, and install one of the following types of Opsware SAS cores or Opsware Satellite:

   - **Standalone Core Installation**: Run the Opsware Installer for the interview and then create the core.

See Chapter 6, “Standalone Core Installation” on page 125 of this guide for more information.

- **Multimaster Core Installation**: Run the Opsware Installer for the interview and then add a core to a multimaster mesh.

  See Chapter 8, “Multimaster Installation” on page 163 of this guide.

- **Satellite Realm Installation**: Run the Opsware Installer for the interview and create an Opsware Satellite in a remote facility.

  See Chapter 9, “Satellite Installation” on page 183 of this guide for more information.

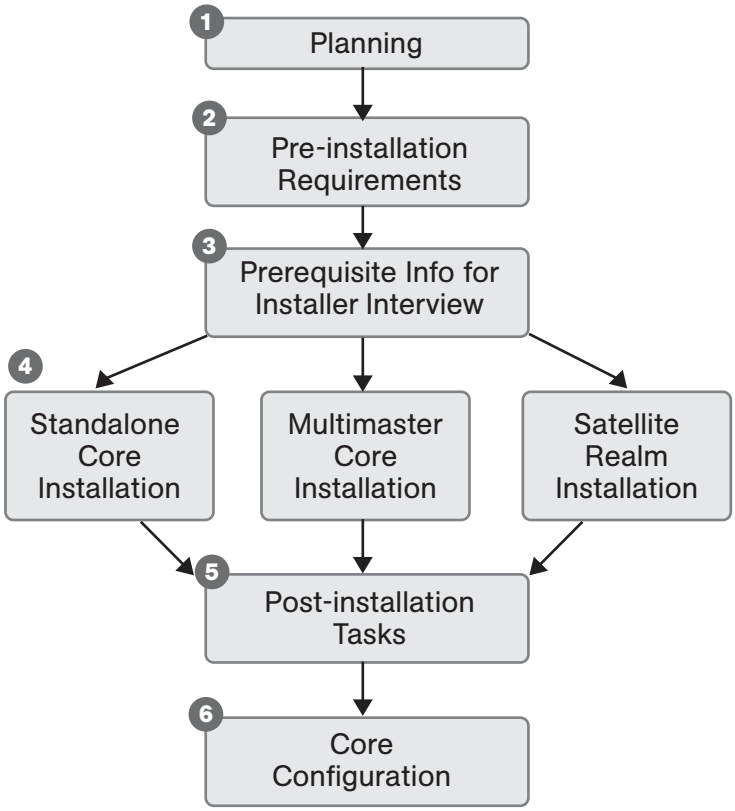
5 **Post-installation Tasks**: Perform hands-on administrative tasks such as configuring the DHCP server in preparation for Opsware OS Provisioning. At the end of this phase, the newly installed Opsware core is up and running.

See Chapter 7, “Standalone Core Post-Installation Tasks” on page 135 of this guide.

6 **Core Configuration**: Configure Opsware SAS, performing tasks such as creating Opsware users, and groups. At the end of this phase, Opsware SAS is ready for operational use by system administrators. See the *Opsware® SAS Administration Guide* for more information.

Figure 4-1 shows the overall process of an Opsware core installation.

Figure 4-1: Opsware Core Installation Process Flow



Installation Checklists

This section discusses the following topics:

- Overall Planning Checklist
- Specific Core Planning Checklist
- Specific Core Requirements Checklist
- Pre-Installation Tasks Checklist
- Post-Installation Tasks Checklist

Overall Planning Checklist

The following checklist summarizes decisions regarding the overall design of your Opsware SAS installation.

Table 4-1: Overall Planning Checklist

OVERALL PLANNING ITEM	ANSWER
How many facilities (data centers) will you manage with Opsware SAS?	
In each of these facilities, how many servers will you manage with Opsware SAS?	
What is your naming convention for the Opsware facility names? (For example, you might use building or city names.)	
Have you taken an inventory of the operating systems and applications on the servers that you will manage with Opsware SAS?	
Which operating systems will you provision (install) with Opsware SAS?	
What applications will you provision (install) with Opsware SAS?	
Which Opsware SAS features will you use?	
What is your schedule for installing Opsware SAS core and for installing agents on the servers to be managed?	
Which of the following Opsware SAS architectures have you chosen? • Standalone • Multimaster mesh • Satellite	

Table 4-1: Overall Planning Checklist (continued)

OVERALL PLANNING ITEM	ANSWER
If you will be using multimaster mesh, how fast is the network connection between the Opsware cores?	
How many cores will you install?	
For each core, in which facility will it reside?	
How many Opsware Satellites will you install?	
For each Satellite, in which remote facility will it reside?	
Which cores will the Satellite communicate with?	
How fast is the network connection between the Satellite and the core?	
Have you drawn a diagram showing the hosts that will run the Opsware core components? If applicable, the diagram should show the network connectivity between multimaster cores and between cores and Satellites.	

Specific Core Planning Checklist

The following checklist summarizes design decisions for a specific Opsware core installation.

Table 4-2: Specific Core Planning Checklist

SPECIFIC CORE PLANNING ITEM	ANSWER
In which facility will this core reside?	
What will be the facility name?	
For the first core, what will be the facility ID and the default customer name?	

Table 4-2: Specific Core Planning Checklist (continued)

SPECIFIC CORE PLANNING ITEM	ANSWER
How many servers will this Opsware core manage?	
Will the Opsware Model Repository use the Oracle software and database installed by the Opsware Installer?	
Will you distribute the Opsware core components across multiple servers?	
What are the host names of the servers on which the core components will be installed?	
For a multiple-server core, have you drawn a diagram that shows which components will run on which servers?	
For a multimaster mesh, will you be using an Opsware Software Repository Replicator?	
For a multiple-server core, will you have multiple instances of the following Opsware components? • Data Access Engine • Opsware Command Center • Media Server • Global File System Server	
Will you deploy a load balancer on multiple instances of the following Opsware components? • Data Access Engine • Opsware Command Center	

Table 4-2: Specific Core Planning Checklist (continued)

SPECIFIC CORE PLANNING ITEM	ANSWER
Will you install the following Opsware components into their own DMZ network? • OS Provisioning Boot Server • OS Provisioning Media Server	
Do you have the necessary licenses for Oracle?	
Have you written your backup and recovery plan for the servers running Opsware SAS?	
Have you contacted your database administrator (DBA)? Your DBA will need to monitor the Oracle database when it goes into production.	
Have you contacted your network administrator? He or she will need to setup host name resolution (/etc/hosts, DNS) before the installation and will run a DHCP configuration tool after the installation.	
Which version of Opsware SAS are you installing?	

Specific Core Requirements Checklist

The following checklist summarizes the technical requirements that must be met before Opsware core installation.

Table 4-3: Specific Core Requirements Checklist

REQUIREMENT	ANSWER
Have the hardware servers on which you will install the Opsware core components (core servers) been racked and stacked?	
Do you have root access to the core servers?	

Table 4-3: Specific Core Requirements Checklist (continued)

REQUIREMENT	ANSWER
Will you be able to mount Opsware SAS DVDs and copy their contents to the core servers?	
Are the core servers running a supported operating system?	
Do the core servers meet the CPU requirements?	
Do the core servers meet the memory requirements?	
Do the core servers meet the disk space requirements?	
Are the servers for an individual core on the same LAN or VLAN? (Multimaster cores must be on separate VLANs.)	
Do the core servers have network connectivity to the servers they will manage?	
Have you verified that Network Information System (NIS) is *not* running on the core servers?	
If you will be using the Network File System (NFS) for Opsware components, such as the Software Repository or Media Server, does the root user have write access over NFS to the directories where the components are to be installed?	
Does the link speed and duplex of core and managed servers match the switch to which they are connected?	
Are the necessary TCP ports open on the core and managed servers?	

Pre-Installation Tasks Checklist

The following checklist summarizes the hands-on tasks you must perform before installing an Opsware core. It is not a series of questions for you to answer, but a series of tasks for you to complete.

Table 4-4: Pre-Installation Tasks Checklist

PRE-INSTALLATION TASK	TASK COMPLETED?
For the servers that will run the Opsware core components (core servers), perform the specific tasks for Linux and Solaris described in the section "Solaris and Linux Requirements" on page 56.	
Set up the host name resolution (/etc/hosts or DNS) for the core servers.	
If network provisioning occurs on a separate network from the Opsware core components, set up DHCP proxying.	
Obtain `mbsacli.exe` and the other utilities required for patches from Microsoft and copy them to a location on your network that is accessible by the Opsware installer.	
Synchronize the system clocks on the core servers with an external Network Time Protocol (NTP) service.	
For a multimaster mesh installation, see the section "Multimaster Installation Prerequisites" on page 166.	
Verify that you have followed the instructions in Chapter 5, "Prerequisites for the Installer Interview".	

Post-Installation Tasks Checklist

The following checklist summarizes the hands-on tasks you must perform after installing an Opsware core. For more information, see the "Post-Installation Tasks" chapter of the *Opsware® SAS Planning and Installation Guide*.

Table 4-5: Post-Installation Tasks Checklist

POST-INSTALLATION TASK	TASK COMPLETED?
Install the Windows Agent Deployment Helper.	
Configure DHCP for Opsware OS Provisioning. You may use the DHCP server included with Opsware SAS or an external DHCP server.	
For Windows OS provisioning, the host name `buildmgr` should resolve on Windows installation clients.	
For Patch Management on Windows NT or 2000, create a silent-installable version of IE 6.0 or later.	
Multimaster mesh: Associate customers with the new facility.	
Multimaster mesh: Update the group permissions for the new facility.	
Multimaster mesh: Verify that the multimaster transaction traffic is flowing between the cores.	

Chapter 5: Prerequisites for the Installer Interview

IN THIS CHAPTER

This section discusses the following topics:

This chapter lists the information needed to complete the Opsware Installer interview and provides information about the installer command line syntax, log files, and Opsware Installer distribution on DVDs.

The Opsware Installer Interview Mode

When you first run the Opsware Installer you must provide certain information about your environment so the installation can be completed. For example:

- Passwords (Opsware Admin, Database Administrator, etc.)
- Service (TNS) Names
- Path names for programs, configuration file, logs
- IP Addresses
- Gateway ports, etc.

You provide this information through a series of prompts from the installer. The specific prompt will vary depending on whether you choose the Simple or Advanced interview mode. All responses you make will be stored on the server in a Response File that is used during the installation and can also be used later during upgrades. After the interview completes, you can either continue the installation using the response file you just created or specify the response file when you install an Opsware core component onto a server later.

This chapter provides detailed information about the installer prompts as well as valid values for the responses.

Opsware Installer Interview Prompts

Before you run the Installer interview, you must gather the information that you will enter when prompted during the interview process. Examples of this information are: the password for the Oracle `opsware_admin` user, the Opsware facility name for the core, and the Opsware authorization domain, etc.

When you run the Opsware Installation script, the Installer prompts you to choose either the **Simple** or **Advanced** interview. If you choose Simple mode, the default values will be used for certain values, for example, passwords for the Oracle database, the `truth` and `spin` user, ports used by the Opsware gateway, among others. In Advanced Mode, you can select values other than the default, giving you finer control.

The tables that follow list the various prompts that you will see when running the Installer interview. Prompts required only for the installation of a multimaster core are indicated by the word **Multimaster** (in bold font). Prompts required only for the advanced mode are designated by the word **Advanced**.

Model Repository Prompts

The Model Repository is the database that stores information about the hardware and software deployed in the operational environment. Most of the Model Repository prompts are for a standalone Opsware core. However, for multimaster mesh cores, you need to provide some additional information The following table lists the Model Repository prompts and the actions associated with them.

Table 5-1: Model Repository Prompts

PROMPT	DESCRIPTION
Enter the service name (aka TNS name) of the Model Repository instance. This is where the Opsware Installer is being run. (Parameter: `truth.servicename`)	Specifies the service name, also known as the alias, for the Model Repository. For an Oracle database created with the Opsware Installer, the service name can be provided during the Opsware Installer interview. For an existing Oracle database, the service name can be determined by looking in the `tnsnames.ora` file on the Model Repository instance. The service name is the value before the first equals sign (=) in the file. The location of this file can vary, so check with your DBA if you are not sure where to look. Source: The DBA who created the Oracle database. Example: `truth.opsware.com`

Table 5-1: Model Repository Prompts (continued)

PROMPT	DESCRIPTION
Enter the service name (aka TNS name) of the Model Repository instance that you will be installing in the new facility. (Parameter: `slaveTruth.servicename`)	**Multimaster**: Specifies the service name, also known as the alias, for the Model Repository of the target core. The service name can be determined by looking in the `tnsnames.ora` file on the Model Repository instance. The location of this file can vary, so check with your DBA if you are not sure where to look. Source: The DBA who created the Oracle database. Example: `truth02.opsware.com`
Enter the SID of the Oracle instance that contains the Data Model Repository. (Parameter: `truth.sid`)	**Multimaster**: Specifies the database system ID (SID) that was set when Oracle was installed on the server where the Model Repository is installed. For an Oracle database created with the Opsware Installer, the SID is `TRUTH`. For an existing Oracle database, you can find out the SID by looking at the `tnsnames.ora` file. The location of this file can vary, so check with your DBA if you are not sure where to look. If you installed the Opsware-supplied Oracle database, you will not be prompted for this parameter. Source: The DBA who created the Oracle database. Example: `DTC05`

Table 5-1: Model Repository Prompts (continued)

PROMPT	DESCRIPTION
Enter the path of the Oracle home. (Parameter: `truth.orahome`)	Specifies the base directory of the Oracle installation that was set when Oracle was installed. For an Oracle database created with the Opsware Installer, ORACLE_HOME is `/u01/app/oracle/product/10.2.0/db_1`. For an existing Oracle database, you can determine the Oracle home directory by logging in as the `oracle user` on the Model Repository server, and checking the value of the `$ORACLE_HOME` environment variable. (For a remote database, this parameter refers to the installation of Oracle Client on the Model Repository server.) If you installed the Opsware-supplied Oracle database, you will not be prompted for this parameter. Source: The DBA who created the Oracle database. Example: `/u01/oracle/product/9.1 or /u01/app/oracle/product/10.2.0/db_1`

Table 5-1: Model Repository Prompts (continued)

PROMPT	DESCRIPTION
Enter the path to the TNS admin directory (where the `tnsnames.ora` file resides). (Parameter: `truth.tnsdir`)	Specifies the directory that contains the `tnsnames.ora` file. This directory must be the same on all servers in the core. For example, since the Data Access Engine requires the `tnsnames.ora` file to connect to the Model Repository, this directory location on the Data Access Engine server must be the same directory location on the Model Repository server. For an Oracle database created with the Opsware Installer, the `tnsnames.ora` file is installed under `/var/opt/oracle`. For an existing Oracle database, the location of the `tnsnames.ora` file can vary, so check with your DBA if you are not sure where to look. If you installed the Opsware-supplied Oracle database, you will not be prompted for this parameter. Source: The DBA who created the Oracle database. Example: `/var/opt/oracle`
Enter the full path to the directory where the export file will be saved. (Parameter: `truth.dest`)	**Multimaster**: Specifies the directory where the database export file will be saved. This directory must exist on the Model Repository server in the source facility. When adding a facility to a multimaster mesh, you must export the Model Repository from the source facility, then copy it to the destination facility. Source: Arbitrary. (However, you must create the directory on the server before you run the Opsware Installer.) Example: `/export/home/core1`

Table 5-1: Model Repository Prompts (continued)

PROMPT	DESCRIPTION
Enter the full path to the directory that contains the export file. (Parameter: `truth.sourcePath`)	**Multimaster**: Specifies the directory on the Model Repository server in the destination facility where the export data file was copied from the source facility. When adding a facility to a multimaster mesh, you must export the Model Repository data from the source facility, then copy it to the destination facility. Source: Arbitrary. (However, the directory must exist on the server and contain the database export file before you run the Opsware Installer on that server.) Example: `/export/home/core2`
Enter the IP address of the device where you are planning to install the Model Repository in the new facility. (Parameter: `slaveTruth.truthIP`)	**Multimaster**: Specifies the IP address of the host on which you will install the Model Repository for the new target core. Source: Arbitrary. Example: `192.168.165.242`
Enter the IP address of the device where you are planning to install the Multimaster Infrastructure Components (vault). (Parameter: `slaveTruth.vaultIP`)	**Multimaster**: Specifies the IP address of the host on which you will install the Multimaster Infrastructure Components for the core. Source: Arbitrary. Example: `192.168.165.242`

Database (Model Repository) Password Prompts

To ensure a secure installation of Opsware SAS, the Opsware Installer prompts you to set passwords for numerous Oracle user accounts that the Opsware components use to interact with one another. The passwords must meet the following standard Oracle criteria:

- The password cannot contain an Oracle reserved word (see Oracle's documentation for a full list).
- The password must be between 1 and 30 characters long.
- The password must start with a letter and use only alphanumeric and underscore (_) characters.

The following table defines the various passwords that you will create and their associated functions.

Table 5-2: Database Password Prompts

PROMPT	DESCRIPTION
Enter database password for the `opsware_admin` user. (Parameter: `truth.oaPwd`)	Specifies the `opsware_admin` password created by your database administrator. `opsware_admin` is an Oracle user that the Opsware Installer uses during installation to perform certain functions. Source: This must be the password that your DBA set for the `opsware_admin` user when setting up the Oracle instance on the server where you will install the Model Repository.
Enter database password for the lcrep user. (Parameter: `truth.lcrepPwd`)	**Advanced**: Sets the password for the `lcrep` database user. The Opsware Installer automatically creates an Oracle user `lcrep`, which Opsware SAS uses internally for running multimaster replication between Opsware cores. Source: Arbitrary. (However, must meet the requirements for Oracle passwords.) Example: `x145_pwd03`

Table 5-2: Database Password Prompts (continued)

PROMPT	DESCRIPTION
Enter database password for the gcadmin user. (Parameter: `truth.gcPwd`)	Sets the password for the `gcadmin` database user. The Opsware Installer automatically creates an Oracle user `gcadmin`, which Opsware SAS uses internally for removing old data from certain tables (referred to as the garbage collection process). Source: Arbitrary. (However, must meet the requirements for Oracle passwords.) Example: `x145_pwd03`
Enter the database password for the truth user. (Parameter: `truth.truthPwd`)	**Advanced**: Sets the password for the he `truth` user. The Opsware Installer automatically creates this Oracle user, which is the main schema owner for the Model Repository. Source: Arbitrary. (However, must meet the requirements for Oracle passwords.) Example: `x145_pwd03`
Enter the database password for the spin user. (Parameter: `truth.spinPwd`)	**Advanced**: Sets the password for the `spin` user. The Opsware Installer automatically creates this database user. Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.) Example: `x145_pwd03` Note: Passwords for the `spin` user must be the same across all the cores in the mesh.

Table 5-2: Database Password Prompts (continued)

PROMPT	DESCRIPTION
Enter the database password for the twist user. (Parameter: `truth.twistPwd`)	**Advanced**: Sets the password for the `twist` user. The Opsware Installer automatically creates this user. Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.) Example: `x145_pwd03`
Enter the database password for the vault user. (Parameter: `truth.vaultPwd`)	**Multimaster**: Sets the Model Repository, Multimaster Component password. This prompt only appears when installing Opsware SAS in multimaster mode. The Opsware Installer automatically creates the `vault` user. The Model Repository, Multimaster Component propagates and synchronizes changes from each Model Repository database to all other Model Repository databases. Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.) Example: `x145_pwd03`

Table 5-2: Database Password Prompts (continued)

PROMPT	DESCRIPTION
Enter the database password for the public views user. (Parameter: `truth.pubViewsPwd`)	**Advanced**: Sets the password for the `public_views` user, which Opsware SAS uses for the Data Center Intelligence (DCI) module (server reporting). The DCI module uses this password when connecting with the Model Repository. The Opsware Installer automatically creates the public views user. If you are using Brio, Crystal Reports, or other data reporting tools with the DCI module, you are asked for the database user password when you log in to those applications so that you have read-only access to the Model Repository data. Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.) Example: `x145_pwd03`
Enter the database password for the AAA user. (Parameter: `truth.aaaPwd`)	**Advanced**: Sets the password for the AAA user, which Opsware SAS uses for the Access, Authentication, and Authorization (AAA) feature. The Opsware Installer automatically creates the AAA user. Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.) Example: `x145_pwd03`
Enter the password to use for DCML exchange tool user. (Parameter: `truth.detuserpwd`)	**Advanced**: Sets the password for the `DETUSER`, which Opsware SAS uses for the DCML Exchange Tool (DET). The Opsware Installer automatically creates the `DETUSER`. Source: Arbitrary. (However, it must meet the requirements for Oracle passwords.) Example: `x145_pwd03`

Opsware Component Password Prompts

This section lists the password prompts for the components other than the Model Repository.

In a multimaster mesh, the following passwords set during the Opsware Installer interview must be the same in all cores belonging to the mesh.

Table 5-3: Component User and Password Prompts

PROMPT	DESCRIPTION
Enter the password for Build Manager user. (Parameter: `twist.buildmgr.passwd`)	**Advanced**: Sets the password for the `buildmgr` user that the `buildmgr` process will use when connecting to and authenticating with the Web Services Data Access Engine. The Opsware Installer automatically creates this user. The password cannot contain spaces or a forward slash (/). Source: Arbitrary. Example: `x145_pwd03`
Enter the password for Integration user. (Parameter: `twist.integration.passwd`)	**Advanced**: Sets the password for the `integration` user that a customer can use to access the SOAP APIs on the Web Services Data Access Engine. The Opsware Installer automatically creates the `integration` user. The password cannot contain a forward slash (/). Source: Arbitrary. Example: `x145_pwd03`

Table 5-3: Component User and Password Prompts (continued)

PROMPT	DESCRIPTION
Enter the password to decrypt cryptographic material. (Parameter: `decrypt_passwd`)	Sets the password to use for decrypting cryptographic material. It cannot contain any spaces. The password must be between 4 and 20 characters long. This password must be the same across all Opsware cores in a multimaster mesh. Source: Arbitrary. Example: `x145_pwd03`
Enter the password to use for admin entry. (Parameter: `cast.admin_pwd`)	Sets the password for the Opsware `admin` user. The password cannot contain any spaces. The Opsware Installer automatically creates the `admin` user. When you first log in to the SAS Web Client in the facility, you log in as the `admin` user. In general, you will *not* need to log in to the directory manager (Netscape Directory Server) by using this user and password unless you need to troubleshoot directory issues. Source: Arbitrary. Example: `x145_pwd03`

Facility Prompts

A facility refers to the collection of servers that a single Opsware core manages. If you are performing a standalone core installation, your deployment is made up of a single facility. Multimaster installations, however, make up two or more facilities: one facility for each core that you install.

Table 5-4: Facility Prompts

PROMPT	DESCRIPTION
Enter the authorization domain (uppercase). (Parameter: `truth.authDom`)	Sets the authorization domain for the initial (default) customer. This value is usually the same as the domain name. It must be uppercase, less than 50 characters, and in domain name format. You must use the same value for every Opsware core in your multimaster mesh. The Opsware Installer only prompts you for this value when you are installing your first, standalone Opsware core. Source: Arbitrary. Example: `XYZ.COM`
Enter the subdomain for this facility (lowercase, no spaces). This is the facility where the Opsware Installer is being run. (Parameter: `truth.dcSubDom`)	Specifies the fully-qualified DNS subdomain where the Opsware core is deployed. This value must be unique for each core in the multimaster mesh. The value is based on the VLAN for the facility in which you are installing the Opsware core. It must be lowercase, less than 50 characters, and in subdomain format. Source: Your network administrator. Example: `dc1.opsware.com`

Table 5-4: Facility Prompts (continued)

PROMPT	DESCRIPTION
Enter the subdomain for the facility you are about to create (lowercase, no spaces). (Parameter: `slaveTruth.dcSubDom`)	**Multimaster**: Specifies the fully-qualified DNS subdomain where the target core is deployed. This value must be unique for each core in the multimaster mesh. The value is based on the VLAN for the facility in which you are installing the target core. It must be lowercase, less than 50 characters, and in subdomain format. Source: Your network administrator. Example: `dc2.opsware.com`
Enter the short name of the facility where the Opsware Installer is being run (uppercase, no spaces). (Parameter: `truth.dcNm`)	Sets the default facility in the source core. Some Opsware SAS processes use this name internally. It must be uppercase, less than 25 characters, and cannot contain spaces or special characters (although underscores are allowed). Dashes are not allowed. Source: Arbitrary. Example: HEADQUARTERS
Enter the short name of the new facility you would like to define (Parameter: `slaveTruth.dcNm`)	Sets the default facility in the target core. Some Opsware SAS processes use this name internally. It must be less than 25 characters, and cannot contain spaces or special characters (although dashes and underscores are allowed). Source: Arbitrary. Example: NORTHSIDE

Table 5-4: Facility Prompts (continued)

PROMPT	DESCRIPTION
Enter the default locale for users of the SAS Web Client. (Parameter: `default_locale`)	Specifies the default locale (language, character sets, and date and time formats) for the Opsware SAS core. Source: In this release, the allowed values are `en` (English) and `ja` (Japanese). Example: `en`
Enter the facility long name. This is the facility where the Opsware Installer is being run. (Parameter: `truth.dcDispNm`)	**Advanced**: Sets the name that displays in the SAS Web Client. It must be unique, less than 50 characters, and cannot include any special characters (< > & * \ ' ?). Source: Arbitrary. Example: Los Angeles Office
Enter the long name for the facility that you are adding to the mesh. (Parameter: `slaveTruth.dcDispNm`)	**Multimaster, Advanced**: Sets the name of the target core that displays in the SAS Web Client. It must be unique, less than 50 characters, and cannot include any special characters (< > & * \ ' ?). Source: Arbitrary. Example: Toronto Office

Table 5-4: Facility Prompts (continued)

PROMPT	DESCRIPTION
Enter the facility ID (number only, less than 1000, with no leading zeros). (Parameter: `truth.dcId`)	Specifies the ID that uniquely identifies a facility. When you install a standalone core, you choose the facility ID during the installer interview. When you install a target core in a multimaster mesh, the facility ID is automatically generated when you add the facility in the SAS Web Client. You specify this automatically-generated ID during the installer interview. Find the target facility ID by logging in to the SAS Web Client at the source facility. Select Opsware Facilities under Environment in the Navigation pane and click the facilities' name. **REQUIREMENT** Opsware facility IDs must be less than 1000. Therefore, you must specify a number for the first facility that is well below 1000 so you can continue to add facilities to your multimaster mesh. Source: Arbitrary for the first facility; set by the Opsware SAS for subsequent facilities. Example: 100

Opsware SAS Feature Prompts

The following prompts are required to configure the OS Provisioning, Software Provisioning, Patch Management, and NAS Integration features in Opsware SAS.

The response to the prompt for the windows utilities directory depends on the steps you performed in “Windows Patch Management Requirements” on page 73.

Table 5-5: Opsware SAS Feature Prompts

PROMPT	DESCRIPTION
Enter the directory that contains Microsoft's qchain.exe, mbsacli20.exe, wusscan.dll, WindowsUpdateAgent20-x86.exe and wsusscan.cab files (Parameter: `windows_util_loc`)	Specifies the directory to which you've copied the Microsoft utilities required for the Patch Management feature on Windows. Source: Arbitrary. (However, this directory must exist on the server where the Software Repository is installed.) Example: `/home/win_util`
Enter the OS Provisioning Boot Server IP or host name. (Parameter: `bootagent.host`)	Specifies the server on which you will install the OS Provisioning Boot Server component. You must provide a valid IP address or host name that can be resolved from the server on which you installed the OS Provisioning Boot Server and the Build Manager. Additionally, the host name must be resolvable by Opsware managed servers for OS provisioning.
Enter the host name or IP of the Build Manager. (Parameter: `boot_server.buildmgr_host`)	Specifies the server on which you will install the OS Provisioning Build Manager. You must provide a valid IP address or host name that can be resolved from the server on which you install the OS Provisioning Boot Server.
Enter the default network speed/ duplex setting for Solaris servers. (Parameter: `boot_server.speed_duplex`)	Sets the default network speed and duplex that will be used by Solaris servers booted from this boot server during Opsware OS provisioning. Valid responses are 100fdx, 100hdx, 10fdx, 10hdx, 100T4, and autoneg. Enter a value without spaces. Source: Arbitrary. Example: `100fdx`

Table 5-5: Opsware SAS Feature Prompts (continued)

PROMPT	DESCRIPTION
Enter the pathname of the Red Hat Linux media. (Parameter: `media_server.linux_media`)	Specifies the path to the Linux OS media on the server on which the Software Repository will be installed. Providing the path to the Linux OS media does not actually copy the media to this host. See the *Opsware® SAS Policy Setter's Guide* for the steps required to set up the media on the Media Server. Source: Arbitrary. (However, this directory must exist on the server where the Software Repository is installed.) Example: `/home/os_media/linux/`
Enter the pathname of the Solaris media. (Parameter: `media_server.sunos_media`)	Specifies the path to the Sun Solaris OS media on the server on which the Software Repository will be installed. Providing the path to the Solaris OS media does not actually copy the media to this host. See the *Opsware® SAS Policy Setter's Guide* for the steps required to set up the media on the Media Server. Source: Arbitrary. (However, this directory must exist on the server where the Software Repository is installed.) Example: `/home/os_media/solaris/`

Table 5-5: Opsware SAS Feature Prompts (continued)

PROMPT	DESCRIPTION
Enter the pathname of the Windows media. (Parameter: `media_server.windows_media`)	Specifies the path to the Microsoft Windows OS media on the server on which the Software Repository will be installed. The OS Provisioning feature exports Windows OS media to SMB clients through a Samba share. Providing the path to the Windows OS media does not actually copy the media to this host. See the *Opsware® SAS Policy Setter's Guide* for the steps required to set up the media on the Media Server. Source: Arbitrary. (However, this directory must exist on the server where the Software Repository is installed.) Example: `/home/os_media/windows/`
Enter the share name to use for the Windows media sharing server. (Parameter: `media_server.windows_share_name`)	**Advanced**: Sets the share name that you want Samba to use to export the Windows OS media. The share name is not case sensitive. Source: Arbitrary. Example: `WINMEDIA`
Enter a password to write-protect the Windows media share. Import_media prompts for this password each time it is run. (Parameter: `media_server.windows_share_password`)	**Advanced**: Sets the root user password, which enables write access to the Windows share. The Opsware Import Media Tool prompts for this password each time it is run. The password cannot contain spaces. Source: Arbitrary. Example: `x145_pwd03`

Table 5-5: Opsware SAS Feature Prompts (continued)

PROMPT	DESCRIPTION
Enter the root directory for the Package Repository. (Parameter: `word_root`)	Specifies the directory where packages are stored on the Software Repository for the Software Provisioning feature. Make sure this directory has sufficient free disk space. By default, packages are stored in the `/var/opt/opsware/word` directory on the Software Repository. Source: Arbitrary. Example: `/var/opt/opsware/word`
Enter the host name or IP address of the NAS server. (Enter "none" if NAS is not installed.) (Parameter: `twist.nasdata.host`)	Specifies the host name or IP address of the server running the Network Automation System (NAS), when your Opsware SAS core includes the NAS Integration feature. If NAS has not been installed for your company, keep the default value, which is `none`, for this prompt. Enter a value without spaces. Source: Your network administrator or Opsware administrator who installed the Network Automation System. Example: `192.168.165.242`

Opsware Gateway Prompts

These prompts are for the IP addresses and ports at which Opsware Gateways can be contacted by core components, Agents, or other Opsware Gateways. The port number must be less than 64001.

Table 5-6: Opsware Gateway Prompts

PROMPT	DESCRIPTION
Enter the port on which the administrative interface for the core gateway will run. (Parameter: `cgw_admin_port`)	**Advanced**: Specifies the port of the Opsware Gateway's administrative interface, which allows you to view the configuration and monitor traffic flow. Source: Arbitrary. Example: 8085
Enter the IP address of the core Opsware Gateway. (Parameter: `cgw_address`)	Specifies the IP address of the core Opsware Gateway. The core gateway is connected directly to the core and communicates directly with core components. Agent Gateways communicate with the core gateway, which then relays the communication to the appropriate core components. Source: Arbitrary. Example: `192.168.165.242`
Enter the port on which core components can contact this gateway to request tunneled connections. (Parameter: `cgw_proxy_port`)	**Advanced**: Specifies the port of the core Opsware Gateway through which components in the same core can request tunneled connections to other components. Source: Arbitrary. Example: 3002

Table 5-6: Opsware Gateway Prompts (continued)

PROMPT	DESCRIPTION
Enter the port on which Agents can contact the core gateway to request connection to core components. (Parameter: `agw_proxy_port`)	Specifies the port of the core Opsware Gateway through which Opsware Agents can request connections to core components. Source: Arbitrary. Example: 3001
Enter the port on which this gateway will listen for connections from other gateways. (Parameter: `cgw_tunnel_listener_port`)	Specifies the port at which this Opsware Gateway will listen for connections from other Opsware Gateways. Source: Arbitrary. Example: 2001

Opsware Global File System Prompts

The following prompts are for specifying IP addresses and directories for the Opsware Global File System.

Table 5-7: Opsware Global File System Prompts

PROMPT	DESCRIPTION
Enter the IP or host name of the nfs server for the Opsware Global File System user home and tmp directories. (Parameter: `ogfs.store.host`)	**Advanced**: Specifies the server from which the storage for the home and tmp directories for the Opsware Global File System will be mounted. Source: Arbitrary. Example: `192.168.198.92`
Enter the absolute path on the nfs server for the Opsware Global File System user home and tmp directories. (Parameter: `ogfs.store.path`)	**Advanced**: Specifies the directory for the storage of the home and tmp directories of the Opsware Global File System. Source: Arbitrary. Example: `/var/opt/opsware/ogfs/export/store`

Table 5-7: Opsware Global File System Prompts (continued)

PROMPT	DESCRIPTION
Enter the IP or host name of the nfs server for the Opsware Global File System where the audit streams will be stored. (Parameter: `ogfs.audit.host`)	**Advanced**: Specifies the IP address of the server where storage for audit streams for the Opsware Global File System will be mounted. Source: Arbitrary. Example: `192.168.165.242`
Enter the absolute path on the nfs server for the Opsware Global File System where the audit streams will be stored. (Parameter: `ogfs.audit.path`)	**Advanced**: Specifies the path for the storage of the audit streams for the Opsware Global File System. Source: Arbitrary. Example: `/var/opt/opsware/ogfs/export/audit`
Enter a comma-separated list of IP address(es) for the devices where the Opsware Global File System (OGFS) is going to be installed in this facility (ip,ip...). (Parameter: `hub.ip`)	Specifies one or more IP addresses of the servers on which to install the Opsware Global File System. Multiple entries are separated by commas. Source: Arbitrary. Example: `192.168.198.92`
Enter the pathname of where you wish the local cache of snapshots and audits to be. This will require a large amount of disk space (4G by default). (Parameter: `spoke.cachedir`)	Specifies the directory where the Global File System service stores snapshots and audits for quick access. By default, the Audit and Remediation features stores snapshots and audits in the directory `/var/opt/opsware/compliancecache`. This cache area is set up to use 4 GB of disk space. Source: Arbitrary. Example: `/var/opt/opsware/compliancecache`

Table 5-7: Opsware Global File System Prompts (continued)

PROMPT	DESCRIPTION
Enter the minimum ID number to use when assigning Unix user IDs to Opsware users. This number must be no less than 1001 and no greater than 90000000, with no leading zeroes. (Parameter: `twist.min_uid`)	**Advanced**: Specifies the minimum UID number that can be used. Unix UIDs are automatically generated for each Opsware user. UIDs will be allocated by counting up from the minimum UID. The default value is 80001. Source: Arbitrary. Example: `80002`
Enter the default Unix group ID number to use when assigning to Opsware users. This number must be no less than 1001 and no greater than 90000000, with no leading zeroes. (Parameter: `twist.default_gid`)	**Advanced**: Specifies the group ID number that is assigned to each Opsware user upon creation. To restrict Opsware users from using certain ports, this group ID has the least amount of network privileges. The default value is 70001. Source: Arbitrary. Example: `70002`

Uninstallation Prompts

The prompts in the following table appear when you are uninstalling an Opsware core.

Table 5-8: Uninstallation Prompts

PROMPT	DESCRIPTION
Do you need to preserve any of the data in this database? (Parameter: `truth.uninstall.needdata`)	Because uninstalling the Model Repository permanently deletes all data in the database, the uninstallation process stops if you answer yes to this parameter, so you have the opportunity to back up the data you would like to preserve. The Opsware Installer does not preserve any data. Example: `y`

Table 5-8: Uninstallation Prompts (continued)

PROMPT	DESCRIPTION
Are you sure you want to remove all data and schema from this database? (Parameter: `truth.uninstall.aresure`)	Because uninstalling the Model Repository permanently deletes all data in the database, the uninstallation process stops if you answer no to this parameter.
Would you like to preserve the database of cryptographic material? (Parameter: `save_crypto`)	If you answer yes, the database of cryptographic material is saved. Otherwise, it is deleted when the uninstallation finishes. Example: `y`
Are you absolutely sure you want to remove all packages in the repository? (Parameter: `word.remove_files`)	If you answer yes, the packages, logs, and cryptographic material for the Software Repository are removed. Example: `y`

Using the Opsware Installer

This section discusses the following topics:

- Installation Media for the Opsware Installer
- Opsware Installer Command Line Syntax
- Installer Interview
- Opsware Installer Logs

Installation Media for the Opsware Installer

Opsware SAS is available on and installable from the following DVD set that contains the scripts for installing, uninstalling, and upgrading components.

- **Product Software**: Contains all packages and scripts necessary to install an Opsware SAS core, including Oracle RDBMS.
- **Agent and Utilities**: Contains packages, (such as the OS Provisioning Boot Agent, Opsware Agents for each operating system, and so on) that need to be uploaded to the Software Repository after the Opsware SAS core has been installed.

- **Satellite Base**: Contains packages and scripts necessary to install the Opsware Gateway and the Software Repository Cache in the Satellite.
- **Satellite Base Including OS Provisioning**: Contains packages and scripts to install Software Repository Caches, Opsware Gateways, and OS Provisioning components in the Satellite.

For the script names, see "Opsware Installer Command Line Syntax" on page 117.

The Product Software DVD and the Agent and Utilities DVD require a DVD drive that supports dual layer.

Copying the DVD to a Local Disk

Opsware Inc. recommends that you copy the contents of the Opsware SAS DVDs to a local disk or to a network share and run the Opsware Installer from that location. When you copy the contents of an Opsware SAS DVD to a local disk or the network, you must create a directory structure that duplicates the structure of the DVD, for example:

```
/opsware_system
```

The path of the directory where you copy the contents of the DVD cannot have spaces.

When you run the Opsware Installer from the common parent directory, `/opsware_system`, the Opsware Installer switches automatically to the directory it needs to complete the part of the installation process that it is currently performing.

Opsware Installer Command Line Syntax

The Opsware Installer is run by using one of the following three scripts:

- `install_opsware.sh` – installs a component
- `upgrade_opsware.sh` – upgrades a component
- `uninstall_opsware.sh` – uninstalls a component

All three of these scripts run with the same command line options, as the following table shows.

Table 5-9: Opsware Installer Command Line Options

OPTION	DESCRIPTION
`-h`	Display the Opsware Installer help for the command line options. To display help during the interview, press `ctrl-I`.
`--resp_file=`*`file`* `(-r` *`file`*`)`	Install an Opsware component, using the values in the specified response file. The installer prompts for the component to install and then runs an interview that only prompts for data missing in the response file. If the response file is incomplete, the installer prompts for the missing information. The installer keeps an inventory of the components that are installed on a given server.
`--interview`	Conduct the installation interview to obtain values for component parameters. At the end of the interview, the installer saves the values in the response file. Usually, you specify this option when you run the Opsware Installer on the host where the Model Repository has been or will be installed. You also specify this option when you have a complete response file but need to run the installer in a different mode, such as converting a standalone core to multimaster. If you specify both the `--interview` and `--resp_file` options, the installer runs the interview, using the values in the response file as the defaults. If you specify no command line options, the installer runs as if you specified the `--interview` option.
`--verbose`	Run the installer in verbose mode.

Installer Interview

The interview prompts you for the mode, either simple or advanced. In the simple mode, the interview does not prompt for parameters that are rarely modified. (Such parameters include the various Oracle passwords used internally by the Opsware components.) If you use the simple mode, the installer will use default values for these parameters. In the advanced mode, the installer prompts for all parameters that are relevant to the type of installation.

The installer validates responses to the interview prompts as you enter them; you are asked to re-enter a value until the installer is able to validate the answer. Some parameters are also revalidated during the actual installation of components. If a response to a prompt cannot be validated at installation, the installer runs a mini-interview.

At any time during the interview, you can press `ctrl-I` to display help for the current prompt.

After all parameters have values, the installer asks if you want to finish the interview. If you want to go back and review or change your answers, press n. If you press y, the installer prompts for the name of the response file in which it will save your answers. (The directory containing the response file must exist.) After saving the file, the installer asks if you would like to continue the installation using the data from the response file. If you press y, the installer displays the Opsware components to install. If you press n, the installer exits.

When you install a core on multiple servers, you should copy the response file to the other servers so that the installations of subsequent components can use the data in the response file.

Opsware Installer Logs

Each time you run the Opsware Installer, it generates the following log file:

`/var/log/opsware/install_opsware/install_opsware.`*`timestamp`*`.log`

If you specify the --verbose option, the following log file is created:

`/var/log/opsware/install_opsware/install_opsware.`*`timestamp`*`_verbose.log`

Some components have supplementary logs that contain additional details about the installation of those components.

See the *Opsware® SAS Administration Guide* for information about the logs for Opsware SAS components.

The installation of the Model Repository creates the following log files:

```
/var/log/opsware/install_opsware/truth/truth_install_number.log
/var/log/opsware/install_opsware/truth/truth_install_number_
verbose.log
```

When you install the first Opsware SAS core, Opsware Inc. recommends as a best practice that you open a second terminal window and issue the following command:

```
tail -f /var/log/opsware/install_opsware/install_
opsware.<date>_verbose.log
```

Where `<date>` is the most recent timestamp.

Obfuscating Cleartext Passwords

During a SAS installation or a SAS upgrade process, some cleartext passwords will be automatically obfuscated and some will not. Some passwords will be obfuscated when Opsware components start up, such as the buildmgr password when the Twist server starts up. Some passwords in certain files will not be obfuscated, such as passwords in the installation logs and Opsware Installer response files.

There are several ways to manually secure cleartext passwords:

- Remove the installation logs.
- Purge sensitive information from the Opsware Installer response files.
- Delete the Opsware Installer response files.

Table 5-10 lists cleartext passwords that are automatically obfuscated and passwords that must be manually secured.

Table 5-10: Cleartext Passwords

CLEARTEXT PASSWORD	FILENAME	AUTOMATICALLY OBFUSCATED	MANUALLY SECURED
admin	`/var/opt/opsware/twist/?DefaultAuthenticatorInit.ldift`	✔	

Table 5-10: Cleartext Passwords (continued)

CLEARTEXT PASSWORD	FILENAME	AUTOMATICALLY OBFUSCATED	MANUALLY SECURED
buildmgr	/var/opt/opsware/crypto/buildmgr/twist.passwd /var/opt/opsware/crypto/occ/twist.passwd /var/opt/opsware/twist/?DefaultAuthenticatorInit.ldift	✔ ✔ ✔	
cleartext admin	/etc/opt/opsware/twist/startup.properties	✔	
detuser	/var/opt/opsware/crypto/twist/detuserpwd /var/opt/opsware/crypto/OPSWhub/twist.pwd	✔ ✔	
integration	/var/opt/opsware/twist/?DefaultAuthenticatorInit.ldift	✔	
root	/var/log/opsware/agent/agent.err		✔
	Opsware Installer response files: /var/opt/opsware/install_opsware/resp /var/opt/opsware/install_opsware/install_opsware* /var/tmp/@* /var/opt/opsware/install_opsware/truth/truth_install_*	 ✔	 ✔ ✔ ✔
spin	/etc/opt/opsware/spin/spin.args	✔	

Table 5-10: Cleartext Passwords (continued)

CLEARTEXT PASSWORD	FILENAME	AUTOMATICALLY OBFUSCATED	MANUALLY SECURED
vault	`/var/opt/opsware/crypto/vault/vault.pwd`	✔	

Removing Installation and Opsware Installer Log Files

Opsware recommends that the installation or upgrade team removes the installation logs, purges sensitive information from the Opsware Installer response files, or deletes the following Opsware Installer response files:

```
/var/opt/opsware/install_opsware/resp

/var/log/opsware/install_opsware/install_opsware*

/var/tmp/@*
```

Opsware also recommends that you delete or remove sensitive information from the response file you saved at the end of the installation interview. The full path and name of the response file is specified by the user at the end of the installation interview.

The Opsware Installer reminds you to remove sensitive log files by displaying the following message at the end of the installation process:

```
################################################################
WARNING: to make sure that no sensitive information is left
on this server, please remove,encrypt or copy to a secure
location the following files and directories:
  -- /var/opt/opsware/install_opsware/resp/*
  -- /var/log/opsware/install_opsware/*
  -- /var/tmp/*.sh
Also, please encrypt or store in a secure location the response
file that you used to install this core.
################################################################
```

Because these files contain information that can be useful for future upgrades (response files) and troubleshooting tasks (log files), Opsware recommends that you move these files to a secure location or encrypt them, instead of deleting them.

Chapter 6: Standalone Core Installation

IN THIS CHAPTER

This section discusses the following topics:

- Standalone Installation Basics
- Standalone Core Prerequisites
- Standalone Core Installation
- Logging in to the SAS Web Client

This chapter provides a general outline of standalone installation, the prerequisites for standalone installation, and the specific steps for performing the installation and logging in to the SAS Web Client.

Standalone Installation Basics

A standalone core manages servers in a single facility. The following steps provide an overview of the standalone installation process. For detailed instructions, see "Standalone Core Prerequisites" on page 127.

1. (Optional) Manually install the Oracle software and create a database. (For details, see Appendix A.)
2. Obtain the Opsware SAS installation DVDs.
3. Run the Opsware Installer (`install_opsware.sh` script) in interview mode. The interviewer prompts you for information about your environment and saves the information in a response file.
4. Run the Opsware Installer and select the Opsware components to install. In this step, the Installer creates the Opsware directories and files on a server. For a single-server installation, you only need to run the Installer once. For a multiple servers, you log on to each server and run the Installer, specifying the components to install. You must install the Opsware core components in the order displayed by the Opsware Installer (see step 14 on page 130).

Decide How to Install the Oracle Database

In an Opsware SAS core, the Model Repository uses an Oracle database. You can choose to:

- Allow the Opsware Installer Install and configure the Oracle database provided by Opsware.
- Create an Oracle database by other means. See Appendix A.
- Use an existing Oracle installation.

The Opsware Installer prompts you for this choice.

The Oracle database installation process consists of two steps:

1. Install the Oracle software.
2. Create the Oracle database (instance).

The Opsware Installer can install an Oracle database configured for use with Opsware SAS. This chapter provides the instructions for installing and configuring Oracle using the Opsware Installer.

If you decide to perform these steps manually, see Appendix A for instructions about installing and configuring your Oracle database manually especially the following sections:

- “Pre-Oracle Universal Installer Tasks” on page 226
- “Manually Creating the Oracle Database” on page 228
- “Post-Create the Oracle RDBMS Tasks” on page 232

The version of the Oracle database that is created using the Opsware Installer is Oracle Database Standard Edition 10.2.0.2. For manual installations, Opsware SAS supports both the Oracle Database Standard Edition and the Oracle Database Enterprise Edition.

Standalone Core Prerequisites

Before you install a standalone core, you must perform the following tasks:

- Plan your Opsware System deployment. When planning for a core, you must decide whether you want to install the core components on a single server or on multiple servers. See Chapter 1, “Opsware SAS Architecture” and “Opsware Core Performance Scalability” on page 49.
- Perform the pre-installation administration tasks, such as configuring the network. See Chapter 3, “Pre-Installation Requirements.”
- Gather information in preparation for the Opsware Installer interview. This information includes the name and ID of the facility for the core. See Chapter 5, “Prerequisites for the Installer Interview.”

Verify that the server for the Opsware Model Repository (Oracle database) meets the prerequisites described in the following sections:

- “Supported Oracle Versions” on page 217
- “Oracle RDBMS Hardware Requirements” on page 218
- ’“Required Operating System Packages and Patches” on page 220

Standalone Core Installation

This section contains step-by-step instructions for running the Opsware Installer (`install_opsware.sh` script).

1. Obtain the Opsware Server Automation System (SAS) installation media.

   See "Installation Media for the Opsware Installer" on page 116, including the recommendation, "Copying the DVD to a Local Disk."

2. On each server where you will install the new Opsware core, mount the Product Software DVD or NFS-mount the directory that contains a copy of the DVD contents.

   The Opsware Installer must have read/write root access to the directories where it installs Opsware components, even NFS-mounted network appliances.

3. On the server where you want to install the Opsware Model Repository, in a terminal window, log in as root.

4. Change to the root directory:
   `cd /`

5. Run the Opsware Installer in interview mode by invoking it with no command-line options:

   ```
   /opsware_system/opsware_installer/install_opsware.sh
   ```

   You must specify the full path to the script. The directory path shown in this step indicates that you copied the Opsware SAS Product Software DVD to a local disk or network share by using the required directory structure.

When you install the first Opsware SAS core, Opsware Inc. recommends that you open a second terminal window and issue the following commands:

```
cd /var/log/opsware/install_opsware/install_opsware
tail -f install_opsware.<date>_verbose.log
```

where `<date>` is the most recent timestamp.

The Opsware Installer displays the following options:

```
Welcome to the Opsware Installer. Please select one of the
following installation options:

1 - Standalone Installation: Standalone Opsware Core
```



```
2 - Multimaster Installation: First Core (convert from
standalone)
3 - Multimaster Installation: Define New Facility; Export
Model Repository
4 - Multimaster Installation: Additional Core
```

6 At the installation options prompt, select the following option:

```
1 - Standalone Installation: Standalone Opsware Core
```

7 At the interview mode prompt, select one of the following options:

```
1 - Simple Interview Mode
2 - Advanced Interview Mode
```

Choose Option 1 to use the default values for many of the configuration parameters. Choose Option 2 to specify all configuration parameters during the interview.

8 At the database configuration option prompt, select the following option:

```
1 - Install Oracle with Opsware
```

For information about installing an Opsware SAS core by using option 2 ("`Use Existing Oracle Database`"), see Appendix A which explains how to install and configure an Oracle database for use with Opsware SAS without using the Opsware installer.

9 Respond to the interview prompts.

The installer displays default values in square brackets [ ].

See "The Opsware Installer Interview Mode" on page 92.

When you run the interview, the paths for the OS provisioning media must already exist on the server where you will install the OS Provisioning Media Server component.

10 Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

If you are satisfied with your answers, press y.

11 Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.stand_single]
```

The response file is a text file that contains the answers you entered during the interview. You can enter the full path and name of the response file or accept the default. In either case, write down the name of the response file. Note that the default file name corresponds to the type of installation.

12 The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file:

```
Would you like to continue the installation using this
response file? (y/n):
```

If you are satisfied with the responses you entered in the interview and you are ready to install the Model Repository now, enter `y` to continue. If you do not want to install the Model Repository now, enter `n`.

13 If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, invoke the Opsware Installer with the `-r` option to specify the response file created by the interview:

```
/opsware_system/opsware_installer/install_opsware.sh -r
<full_path_to_response_file>
```

14 At the components prompt, select one or more components to install:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Oracle RDBMS for SAS
2 ( ) Model Repository (truth)
3 ( ) Data Access Engine (spin)
4 ( ) Command Engine (way)
5 ( ) Software Repository (word)
6 ( ) Opsware Global Filesystem Server (OGFS)
7 ( ) Opsware Command Center (OCC)
8 ( ) OS Provisioning Media Server
9 ( ) OS Provisioning Build Manager
10 ( ) Opsware Gateway
11 ( ) OS Provisioning Boot Server
```



```
Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection:
```

You must install the components in the order that they are listed. For example, you must install the Model Repository before the Data Access Engine.

If you are installing all of the components on a single server, enter `a` for all. If you do not select `a`, you must run the Opsware Installer again (specifying the response file) and select the remaining components. (If you are installing the components on multiple servers, go to step 15.)

For some of the components, such as the OS Provisioning Build Manager, the Installer interview prompts you for the IP address or host name. Be sure to install these components on the host that you indicated during the interview.

15 If you are installing the components on multiple servers, follow the instructions in this step. (If you are installing the components on a single server, skip this step.)

- Copy the response file generated by the installer interview to all other servers in this core.
- After you install the Model Repository, copy the Oracle `tnsnames.ora` file from the server with the Model Repository to the other Opsware core servers. Make sure that the path for the file (`/var/opt/oracle/tnsnames.ora`) is the same on all core servers. For more information, see "tnsnames.ora File Requirements" on page 233.
- On each server in this core, run the Opsware Installer with the `-r` option, as shown in step 13. Select and install the remaining components from the menu shown in step 14.
- For the Model Repository, enter `y` when the installer asks whether you want to generate a new database of cryptographic material. Copy the database of cryptographic material and the Unix tar file Gzipped from the following directory to every Opsware core server:

  `/var/opt/opsware/crypto/cadb/realm/opsware-crypto.db.e`

  `/var/opt/opsware/crypto/cadb/realm/opsware-crypto.tgz.e`

  The database of cryptographic material and the Tar file Gzipped must be copied to the same directory and file names on every Opsware core server. The directory and database need to be readable by the root user.

- If the Model Repository or Boot Server exist on a server with no other Opsware components installed on it, you must install an Opsware Agent on that server. See the *Opsware® SAS User's Guide: Server Automation* for instructions.

16 (Optional) If you are distributing the core components across multiple servers, you can install additional instances of the following components:

- Data Access Engine

  If you install more than one Data Access Engine, then you must perform the procedure described in "Reassigning the Data Access Engine to a Secondary Role" in the *Opsware® SAS Administration Guide*.

- OS Provisioning Media Server
- Opsware Command Center
- Opsware Global File System Server (OGFS)

  You can install multiple instances of the OGFS when you install an Opsware core. To do so, during the Opsware Installer interview, specify the IP addresses of the servers that you plan to install the OGFS on.

  To install additional instances of the OGFS to an existing core, you must perform manual steps. See Chapter 7, "Adding Instances of the OGFS to a Core" on page 160 of this guide for more information.

17 On the server where you installed the Software Repository, mount the Agent and Utilities DVD or NFS-mount the directory that contains a copy of the DVD contents.

The Opsware Installer must have read/write root access to the directories where it installs Opsware components, even NFS-mounted network appliances.

18 In a terminal window, log in as root and change to the root directory:

```
cd /
```

19 Invoke the Opsware Installer with the `-r` (response file) option. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r
/usr/tmp/oiresponse.standalone
```

You must specify the full path to the script, where `/opsware_system/` is a variable that refers to where the media is installed on the system, such as `/opsware_media/disk1`. The directory path in the preceding command indicates that you copied the Opsware SAS Agent and Utilities DVD to a local disk or network share using the required directory structure.

You should run the Opsware Installer with the response file that you created when you installed the standalone core.

The Opsware Installer displays following options:

```
Welcome to the Opsware Installer.
Please select the components to upgrade.
1 ( ) Software Repository - Content (install once per mesh)
2 ( ) Add OS Provisioning Stage2 Images to Software
Repository
Enter a component number to toggle ('a' for all, 'n' for
none.
```

20 At the install prompt, select options 1 and 2:

```
1 ( ) Software Repository - Content
2 ( ) Add OS Provisioning Stage2 Images to Software
Repository
```

21 Follow the instructions in the following section, “Logging in to the SAS Web Client” on page 133.

22 Follow the instructions in “Standalone Core Post-Installation Tasks” on page 135.

Logging in to the SAS Web Client

After you install an Opsware SAS core, you should be able to log in to the SAS Web Client.

To use the SAS Web Client, your browser must be configured in the following manner:

- The browser must accept cookies and be able to use Java.
- The browser must support SSL and should provide 128-bit encryption (recommended).
- Using a pop-up blocker might prevent some functions from working correctly. Either disable the pop-up blocker completely or use the supported browser’s native pop-up blocking function instead of a third-party product.

To log in to the SAS Web Client, perform the following steps:

1 In a web browser, enter the following URL:

```
https://<occ_host>
```

The `<occ_host>` is the host name or IP address of the server on which you installed the Opsware Command Center component.

2 Follow the browser’s instructions for installing the security certificate.

3. When the SAS Web Client prompts you for the user name and password, enter `admin` for the user name. For the password, enter the value for the `cast.admin_pwd`, which you specified during the Installer interview.

4. Create a new user by using the Users & Groups page under Administration. For the Group Membership, select Opsware System Administrators.

   See the *Opsware® SAS Administration Guide* for information about creating Opsware users.

5. Log in to the SAS Web Client as the user you created in the previous step. Run the Opsware System Diagnosis by clicking System Diagnosis under Administration in the navigation panel.

   See the *Opsware® SAS Administration Guide* for information about the procedures for running the system diagnosis tool.

6. Log in to the SAS Web Client as the `admin` user again. Create a new user and for the Group Membership, select Advanced Users.

7. Log in to the SAS Web Client as the user you created in the previous step. Exercise the different Opsware System functions. Click the links in the Navigation pane and open the windows on the home page.

Chapter 7: Standalone Core Post-Installation Tasks

IN THIS CHAPTER

This section discusses the following topics:

- Unattended Installation of the SAS Client Launcher
- Opsware Discovery and Deployment
- NAS Integration
- DHCP Configuration for OS Provisioning
- Additional Network Requirements for OS Provisioning
- Windows Patch Management Tasks
- Support for Redhat Network Errata and Channels
- Opsware Global File System Tasks

This chapter describes system administration tasks that you must perform after installing a core. It provides instructions and requirements for an unattended installation of the SAS Client Launcher, and for setting up and configuring your system for Opsware Discovery and Deployment, OS provisioning, Patch Management, NAS integration, and the OGFS.

The SAS Client

The SAS client is a powerful Java client for the Opsware Server Automation System. It provides the look-and-feel of a Microsoft Windows desktop application with the cross-platform flexibility of Java. Once installed, you can access the SAS Client from any core in your mesh.

To access the SAS Client for the first time, you must access the SAS Client installer from the SAS Web Client. You only need to install the SAS Client once. From the SAS Client home page, click on Install SAS Client. This will download the SAS Client installer

See the *Opsware SAS User's Guide: Server Automation* for more information.

Unattended Installation of the SAS Client Launcher

You can also set up unattended installation of the SAS Client using a command line argument that launches the installer.

To begin an unattended installation, you invoke the installer using the `-q` argument, which causes the installer to perform the installation as if you had accepted all default settings.

For example, execute the following command on the server where you want to install the launcher:

```
opswclientinstaller_windows_1_0.exe -q
```

By default, the launcher is installed in the following directory:

```
C:\Opsware
```

If you want to install the launcher in another directory, specify the `-d` option, as in the following example:

```
opswclientinstaller_windows_1_0.exe -q -dir C:\Opsware_
Launcher
```

See the *Opsware® SAS User's Guide: Server Automation* for information on how to use the SAS Client Launcher.

Opsware Discovery and Deployment

With the Opsware Discovery and Deployment (ODAD) feature, you can use the SAS Client to install Opsware Agents on servers.

Enabling the ODAD Feature for Unix Servers

Enabling the ODAD feature for Unix servers does not require that you perform additional set up steps. When you run the Opsware Installer, it automatically installs all required software to use the ODAD feature with Unix servers.

However, before you use the ODAD feature to open remote terminal sessions on unmanaged Unix servers, verify that the following requirement has been met.

On the server with the Agent Gateway, the `telnet`, `rlogin`, and `ssh` clients must reside in either the `/bin`, `/usr/bin`, or `/usr/local/bin` directory. If the client resides in a different directory, create a symbolic link in `/usr/local/bin` to the actual location of the client.

Enabling the ODAD Feature for Windows Servers

Before you can use the ODAD feature to deploy Opsware Agents to Windows servers, you need to install additional software on a Windows server and configure the Opsware Gateway as described in "Installing the Windows Agent Deployment Helper" on page 137.

Installing the Windows Agent Deployment Helper

Before using the ODAD feature to install Agents on Windows servers, you must install the Windows Agent Deployment Helper.

You need to install only one Windows Agent Deployment Helper for each Opsware core. You cannot install a Windows Agent Deployment Helper in an Opsware Satellite.

To install the Windows Agent Deployment Helper, perform the following steps:

1. Obtain a Windows server on which you can install the Windows Agent Deployment Helper. This server must be running a 32-bit version of Windows 2000, Windows 2003, or Windows XP. (Windows 64-bit operating systems are not supported.)

   On this Windows server, install an Opsware Agent with the command-line utility. For instructions, see "Opsware Agent Utilities" in the *Opsware® SAS User's Guide: Server Automation*.

2. Log in to the SAS Client. See the *Opsware® SAS User's Guide: Server Automation* for information.

3. From the Navigation pane, select **Devices ➤ All Managed Servers**.

4. From the Content pane, select the Windows server on which you installed the Opsware Agent.
5. From the **Action** menu, select **Attach ➤ Attach Software Policy**. The Attach Software Policy window appears.
6. From the list of software policies, select Windows Agent Deployment Helper. (By default, the Remediate Servers Immediately option is selected. Do not deselect this option.)
7. Click **Attach**. The Remediate window appears.
8. Complete the tasks to remediate the server with the Windows Agent Deployment Helper policy. See the *Opsware® SAS User's Guide: Application Automation* for the steps to remediate a server with a software policy.
9. Restart any running SAS Clients.

   The restart is needed because the SAS Client caches information about the Windows Agent Deployment Helper.
10. Log in as `root` to the server with the core Gateway. With a text editor, open the following file:

    ```
    /etc/opt/opsware/opswgw-cgw0-<facility>/opswgw.properties
    ```
11. Locate the following line:

    ```
    #opswgw.IngressMap=${NETBIOSHELPERIP}:NETBIOS
    ```
12. Uncomment the line, and replace `${NETBIOSHELPERIP}` with the IP address of the server where you installed the Windows Agent Deployment Helper. For example:

    ```
    opswgw.IngressMap=192.168.165.242:NETBIOS
    ```
13. Restart the core Gateway with the following command:

    ```
    /etc/init.d/opsware-sas restart opswgw-cgw0
    ```

Details: Agent Deployment Helper Setup for Disabled Administrator Account

When the Windows Administrator account is disabled on a Windows server, you must perform the following additional setup steps for installing the Agent Deployment Helper.

1. Log in as root to a server running an Opsware SAS component.
2. Change directories to the following directory:


```
cd /opt/opsware/oi_util/bin/
```

3 Enter the following command to run the `shared_script_util.sh` script:

```
./shared_script_util.sh modify adt_deploy_agents.bat -U
ACCOUNT_NAME -p agentDeployment.deployAgent -e -c "Change
user name"
```

Where `ACCOUNT_NAME` is the name of the account you want the script to run as.

4 (Optional) Enter the following command to review the current script settings:

```
./shared_script_util.sh showpolicy adt_deploy_agents.bat
```

You will see the following output, except that the `USER` line should contain the name of the account you just set.

```
PTY 0
USER Administrator
EXEMPT
PERM agentDeployment.deployAgent
```

NAS Integration

To set up the NAS Integration feature, you must change configuration settings in NAS and in SAS, run diagnostics for NAS topology data, and set up user permissions.

To set up NAS Integration, you must have Opsware Network Automation System (NAS) 6.1 or later installed.

You can reset the NAS host name if the SAS Client is not communicating with (cannot find) the NAS server as it is currently defined.

When setting up the NAS Integration feature, the Opsware NAS core can use an existing Opsware Gateway installed for Opsware SAS, but SAS cannot use an existing Opsware Gateway installed for NAS. In this release, NAS must be configured to use the Opsware Gateway that is installed by the Opsware SAS installer.

NAS Integration Port Requirements

Before you configure the NAS Integration feature, make sure that SAS and NAS can communicate with each other over the following ports.

NAS to SAS Port Requirements:

- NAS needs to access port 1032 on the server running the Opsware SAS Web Services Data Access Engine component. By default, the Opsware SAS Web Services Data Access Engine component listens on port 1032.

SAS to NAS Port Requirements:

- For the Global Shell feature in Opsware SAS to display data about network devices, Opsware SAS needs access to port 8022 (Unix-based NAS Servers) and 22 (Windows-based NAS Servers).
- The NAS API uses Java RMI to connect to the NAS server. When setting up NAS Integration for Opsware SAS, SAS uses the NAS API for the integration. RMI/JRMP requires that the following ports are open:
  - JNDI (typically 1099)
  - RMI (typically dynamic)
  - RMI Object (typically 4444)

  See the *Opsware® NAS User's Guide* for information about how to set up these port requirements to access the NAS API through a firewall.

Configuring for NAS Integration

To set up the NAS Integration feature, you must change the following two configuration settings, first one in NAS and then one in SAS:

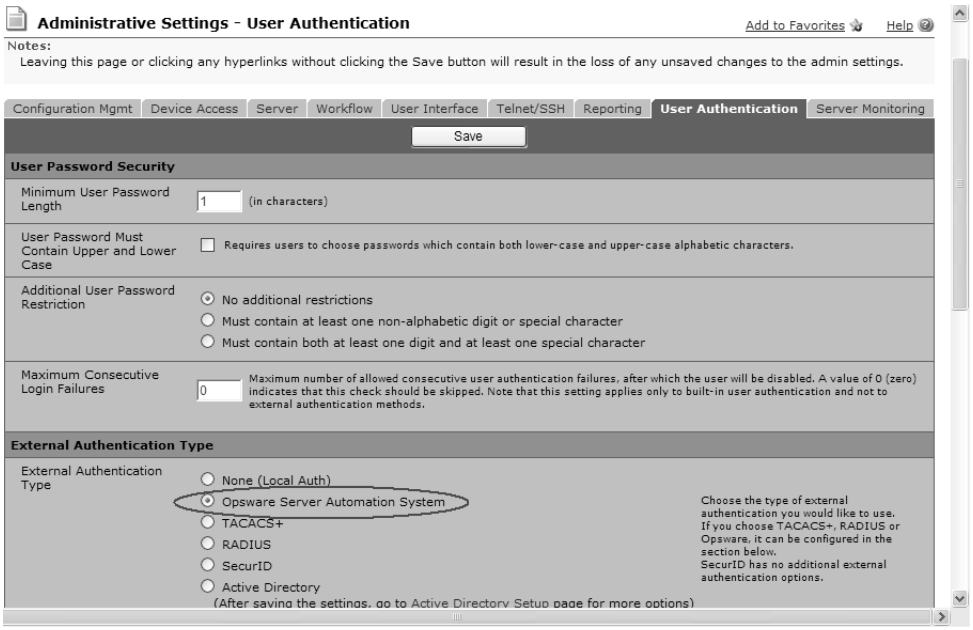
NAS Configuration

To change the configuration setting in NAS, perform the following steps:

1. Log in to Opsware NAS.

2 Select **Admin ➤ Administrative Settings ➤ User Authentication** to display the Administrative Settings – User Authentication page.

Figure 7-1: External Authentication Type in NAS



3 In the External Authentication Type section, select Opsware Server Automation System.

Figure 7-2: Opsware Server Automation System Authentication

4 Complete all fields in the Opsware Server Automation System Authentication section.

NAS uses the Twist Username and Twist Password when it gathers layer 2 data. NAS looks for the server interface information by MAC address, using that user's permissions. The user must have read access to server information.

5 Click **Save** to save your configuration change.

See the *Opsware® NAS User's Guide* for more information on configuration.

SAS Configuration

If the NAS server name was not entered during SAS installation, you must add the `twist.nasdata.host=<hostname>` setting in the `twist.conf` file in `/etc/opt/opsware/twist/twist.conf`. See the *Opsware® SAS Administration Guide*.

After you make this configuration change, you must restart NAS and the Twist server.

If the NAS server is running on Windows, you must change the port setting `nas.port=8022` to `nas.port=22` in the `hub.conf` file in `/etc/opt/opsware/hub/hub.conf`. A default install on Windows runs the proxy SSH/Telnet servers on port 22/23 (instead of port 8022/8023. See the *Opsware® NAS User's Guide* for more information on NAS servers.

After you make this configuration change, you must restart the OGFS server.

Configuring SAS Integration with CiscoWorks NCM

If you are deploying Opsware SAS with CiscoWorks NCM 1.2, you must make configuration changes. Some CiscoWorks NCM deployments (where CiscoWorks LMS is co-resident with NCM) will use non-standard ports that affect integration with SAS.

To determine which changes you will need to make, perform the following steps:

1 Log in to your NAS server.

2 Open `<NAS_install_dir>/server/ext/jboss/server/default/deploy/tomcat4-service.xml`.

3 Search for '`scheme="https`"'.

4 If the port is 443, then you can skip ahead to step 12; otherwise, note the port being used (such as 9443) and then continue to step 5.

5 In the SAS Client, from the **Tools** menu, select **Options**.

6. In the Set Options window, select Opsware NAS.
7. In the Host field, append :<port> to the hostname, where <port> is the port number found in step 4, such as:

   `mycore.opsware.com:9443`
8. The following warning will appear: "General.Host: must be a valid host string." Ignore this warning. Close the Set Options window.

   (Steps 5-8 must be performed for every user of the SAS Client.)
9. Log in to the SAS core server where the Data Access Engine is installed.
10. Open the `/opt/opsware/twist/twist.sh` file and change this:

    `https://$NASHOST/tcdocs/truecontrol-client.jar`

    to this (assuming that 9443 was the port you found in step 4):

    `https://${NASHOST}:9443/tcdocs/truecontrol-client.jar`
11. Restart the Twist server:

    `/etc/init.d/opsware-sas restart twist`

    (You will need to perform steps 9-11 for each Twist server installation.)
12. Log in to Opsware NAS.
13. Select **Admin ➤ Administrative Settings ➤ Telnet/SSH** to display the Administrative Settings - Telnet/SSH page.
14. In the SSH Server section, locate the SSH Server Port.
15. If the port is 8022, then you are finished; otherwise, note the port being used (such as 9022) and then continue to step 16.
16. Log in to the SAS core server where the Hub is installed.
17. Open the `/etc/opt/opsware/hub/hub.conf` file and change the value for nas.port to the port you found in step 15. For example:

    `nas.port=9022`

Topology Data

To continue setting up the NAS Integration feature, you must also run the NAS Topology Data Gathering and NAS Duplex Data Gathering diagnostics in NAS. See the *Opsware® SAS User's Guide: Server Automation* and the *Opsware® NAS User's Guide*.

User Permissions for NAS Integration

Access permissions for the NAS Integration feature are based on two separate databases: a NAS database and a SAS database. NAS uses its own database for authorization. SAS uses a different security mechanism for authorization. However, all authentication (for both NAS and SAS) is processed by SAS.

When NAS is configured to use SAS authentication, it tries to authenticate against SAS first. If NAS fails to authenticate against SAS, it falls back to the NAS database. If there is an account in the NAS database, the fallback is only allowed if that user is configured to allow fallback authentication. See the *Opsware® NAS User's Guide* for more information on NAS authentication.

When a new user is authenticated through SAS, an account is created in NAS. The account is placed in the Default User Group that was specified when SAS authentication was enabled in the Administrative Settings in NAS. This user group, which is configurable, controls the default permissions that the system administrator has assigned to SAS users.

You must have a set of permissions to view servers and network devices, and Twist server configuration changes. To obtain these permissions, contact your Opsware administrator, or for more information, see the *Opsware® SAS Administration Guide*.

DHCP Configuration for OS Provisioning

The Dynamic Host Configuration Protocol (DHCP) specifies how to assign dynamic IP addresses to servers on a network. Opsware OS Provisioning uses DHCP to allow network booting and configuration of unprovisioned servers in the Server Pool. DHCP is also used to configure networking on newly provisioned servers that have not been assigned a static network configuration.

For OS provisioning, you may use either the DHCP server included Opsware SAS, an existing ISC DHCP server, or the MS Windows DHCP server. The instructions for configuring these various DHCP servers are in the following sections:

- Configuring the Opsware DHCP Server for OS Provisioning
- Configuring an Existing ISC DHCP Server for OS Provisioning
- Configuring the MS Windows DHCP Server for OS Provisioning
- Configuring the Opsware and MS Windows DHCP Servers for OS Provisioning

DHCP Software included with the Opsware Boot Server

When you install the Opsware Boot Server, the Opsware Installer also installs the following:

- **dhcpd**: An Internet Software Consortium DHCP server (ISC dhcpd).
- **dhcpd.conf**: A default DHCP server configuration file, read by the `dhcpd` server.
- **dhcpdtool**: The Opsware DHCP Network Configuration Tool which allows you to modify the `dhcpd.conf` file.

Opsware DHCP Server (dhcpd)

The DHCP server provides service to two types of networks:

- **Local networks**: Networks that are attached directly to the network interfaces of the host running the DHCP server. No special network configuration is needed to support local networks.
- **Remote networks**: Networks that are not directly attached to the DHCP server host. A router sits between the DHCP server host and the remote networks. For remote networks, a DHCP proxy (sometimes called IP helper) must be configured on each remote network to relay DHCP packets to the DHCP server host.

A DHCP proxy is not provided with Opsware SAS and instructions for setting one up are beyond the scope of this document. DHCP proxy functionality is often included in modern routers. Check with your network administrator or router vendor.

Log messages that the DHCP server produces are sent to the standard Unix syslog process with the daemon facility. Consult your vendor documentation on how to configure and view syslog messages.

See “Starting and Stopping the Opsware DHCP Server” on page 149.

Opsware dhcpd.conf File

The `dhcpd.conf` file provides the necessary parameters to support network booting of Sun hardware (a DHCP-capable PROM is required) and x86 hardware (a PXE-compatible system is required).

For x86 hardware that does not support PXE, the server can be booted from a floppy (Windows) or CD (Linux). When a boot floppy or CD is used, the DHCP server still provides network configuration information to the host.

The DHCP configuration file is `/etc/opt/opsware/dhcpd/dhcpd.conf`. In most cases, you will modify this file by running the DHCP Network Configuration Tool. For some advanced configurations (as noted in the following section), you may need to modify the file with a text editor. Documentation on the DHCP configuration file is available at the ISC web site `www.isc.org`.

The DHCP leases file is `/var/opt/opsware/dhcpd/dhcpd.leases`. This file should not need editing.

Opsware DHCP Network Configuration Tool (dhcpdtool)

The DHCP Network Configuration Tool is a menu-driven, terminal-based utility that enables you to customize the `dhcpd.conf` file for common local and remote network configurations. The tool prompts you for network information needed to configure DHCP for each OS provisioning network. Using the DHCP Network Configuration Tool simplifies configuration of the DHCP server and ensures that the DHCP configuration contains the options that are needed for the OS Provisioning feature to function properly.

If you need to configure the network for Opsware OS Provisioning to support less common configurations, you must modify the `dhcpd.conf` file with a text editor. Less common configurations include dual-interfaces with split-horizon DNS requirements, private build networks, and static NAT. Contact Opsware Support for more assistance.

Additionally, in some environments, multiple IP networks (layer 3) are layered on top of a single VLAN (layer 2). While this configuration is supported by the ISC DHCP server, generally such a topology requires careful consideration to work properly with DHCP. Therefore, the DHCP Network Configuration Tool can only configure a single IP network per VLAN.

The man pages for the DHCP Network Configuration Tool are installed in `/opt/opsware/dhcpd/man` on the Boot Server. They are also available at the Opsware Support web site.

Required Information for the Opsware DHCP Network Configuration Tool

Before you use the DHCP Network Configuration Tool to configure an OS provisioning network, you need the following information:

- The range of IP addresses that are assigned dynamically by the DHCP server. For example, 192.168.0.11 - 192.168.0.20 might be used to configure a pool of 10 addresses.

Each of these IP addresses must resolve to a host name on the DNS server.

- The IP addresses of one or more DNS servers. The servers must be able to resolve the standard required Opsware DNS entries. The DNS servers do not need to be on the same network that is being configured.
- A default DNS domain. This domain must include the standard, required Opsware DNS entries. For example, if the default DNS domain is `example.org`, then there must be an entry `spin.example.org` that can be resolved by the DNS servers.

If you are going to configure a remote network with the DHCP Network Configuration Tool, you will also need to provide the following information:

- The network address and size (netmask or bits). For example, 192.168.0.0/255.255.255.0 or 192.168.0.0/24. Both specify a network range of 192.168.0.0 - 192.168.0.255.
- The network gateway or default router, for example, 192.168.0.1.

Configuring the Opsware DHCP Server for OS Provisioning

The DHCP Network Configuration Tool is installed with the Opsware Boot Server. Perform the following steps to configure networks for OS provisioning:

1. Log in as root to the server running the Opsware Boot Server.
2. Make a backup copy of the configuration file with the following commands:

   ```
   cd /etc/opt/opsware/dhcpd
   cp dhcpd.conf dhcpd.conf.orig
   ```

3. Run the DHCP Network Configuration Tool with the following command:

   ```
   /opt/opsware/dhcpd/sbin/dhcpdtool
   ```

   The following DHCP Network Configuration Tool main menu appears:

Example: DHCP Network Configuration Tool Main Menu

```
Opsware DHCP Network Configuration Tool

 a)dd a new network.
 e)xit.

Choice [a, e]:
```

4 To add a new network, enter `a` at the preceding prompt.

The following menu to add local or remote networks appears:

Example: Menu to Add Local or Remote Networks

```
Opsware DHCP Network Configuration Tool

You may view/edit/delete one of the currently configured
network(s):

 1) 192.168.164.0/28
 2) 192.168.165.128/28

Or

 a)dd a new network.
 e)xit.

Choice [1..2, a, e]: a:
```

5 To configure the DHCP service on the local network, enter `l` at the preceding prompt. Local networks are detected automatically and displayed.

Or

To add a remote network, enter `r` at the preceding prompt.

6 If you are adding a local network, you need to enter the IP addresses or host names of the DHCP range and the DNS servers.

In the following example, note that the IP addresses are separated by a comma and a space.

Example: Local Network Configuration

```
Opsware DHCP Network Configuration Tool

Editing DHCP information for 192.168.8.0/23 (255.255.254.0)

All values which prompt for an address accept either a IP or a
hostname.

Enter the DHCP Range (start address, stop address)
: 192.168.8.20, 192.168.8.29
Enter the DNS server(s) (comma separated)
: 192.168.2.25, 192.168.2.28
```



```
Enter the DNS domain: opsware.com
```

7 If you are adding a remote network, supply information for the network address, size, and gateway. See the following example:

Example: Remote Network Configuration

```
Opsware DHCP Network Configuration Tool

All values which prompt for an address accept either a IP or a
hostname.

Enter network/netmask or network/bits: 192.168.10.0/24
Enter the network gateway: 192.168.10.1
Enter the DHCP Range (start address, stop address)
: 192.168.10.51, 192.168.10.59
Enter the DNS server(s) (comma separated)
: 192.168.2.25, 192.168.2.28
Enter the DNS domain: opsware.com
```

8 If the displayed information is correct, enter `k` to keep the network and return to the main menu.

9 At the main menu, to save the information you have entered, enter `s`.

Or

To edit a configured network, enter the corresponding integer and go back to step 3.

Or

To add more networks, enter `a` and go back to step 3.

10 To exit the DHCP Network Configuration Tool, enter `e`. You are prompted to start (or restart) the DHCP server process.

11 To start (or restart) the DHCP server process, enter `y`. The DHCP Network Configuration Tool displays diagnostic output as part of its startup.

Starting and Stopping the Opsware DHCP Server

To start the DHCP server process, enter the following command on the server running the Opsware Boot Server:

```
/etc/init.d/opsware-sas start dhcpd
```

To stop the DHCP server process, enter the following command on the server running the Opsware Boot Server:

```
/etc/init.d/opsware-sas stop dhcpd
```

Configuring an Existing ISC DHCP Server for OS Provisioning

You may use an existing ISC DHCP server for OS provisioning instead of the DHCP server included with Opsware SAS. An existing ISC DHCP server will work with the provisioning of PXE 2.0 clients, but not with older clients such as PXE 0.99 or 1.0. (These older PXE clients have old PROMS and a PXE bootstrap floppy made with `rbfg.exe`.) The following instructions apply to recent versions of an ISC DHCP server, such as version 3.02rc3.

To configure an existing ISC DHCP server, perform the following steps:

1. On the server where you installed the Opsware Boot Server, you should prevent the Opsware DHCP server from running.

   On Linux, enter the following command:

   ```
   chkconfig --level 345 dhcpd off
   ```

   On Solaris, enter the following commands:

   ```
   rm /etc/rc2.d/S90dhcpd
   rm /etc/rc0.d/K30dhcpd
   ```

2. Ensure that the configuration file for the existing ISC DHCP server has the entries shown in: "**Example: Configuration File Entries for an Existing ISC DHCP Server**" on page 151.

   The example is a snippet of the `dhcp.conf` shipped with Opsware SAS, with the addition of `next-server`. This addition tells the PXE client to look for the `tftpserver` on the Opsware core, not on the existing DHCP server.

3. If you copy and paste the example, change all of the IP addresses (`1.2.3.4`) to the IP address of your core.

4. Ensure that the DHCP scope for the systems to be provisioned is set up with the required details, such as the DNS server, netmask, default router, DNS domain, and so forth.

5. Restart the existing ISC DHCP server.

Example: Configuration File Entries for an Existing ISC DHCP Server

```
#
# declare OPSW site options
#
option space OPSW;
#
# DANGER WILL ROBINSON - if you change the codes for these
# options, you'll need to also edit them in the param-request-
# lists appearing below. Note that in the pxeclient section, you
# need to specify the values in hex, not in decimal. Also, these
# values are burned into a couple other files you'll need to
# edit as well:
# /opt/opsware/boot/tftpboot/pxelinux.cfg/default
# /opt/opsware/boot/jumpstart/Boot/etc/dhcp/inittab
# /opt/opsware/boot/jumpstart/Boot/etc/default/dhcpagent
#
option OPSW.buildmgr_ip code   186 = ip-address;
option OPSW.buildmgr_port code 187 = unsigned integer 16;

#
# define OPSW site options
#
site-option-space "OPSW";
option OPSW.buildmgr_ip   1.2.3.4;
 option OPSW.buildmgr_port 8017;

#
# declare SUNW jumpstart vendor options (Sun recommended naming)
#
option space SUNW;
option SUNW.SrootIP4 code 2  = ip-address;
option SUNW.SrootNM  code 3  = text;
option SUNW.SrootPTH code 4  = text;
option SUNW.SbootFIL code 7  = text;
option SUNW.SinstIP4 code 10 = ip-address;
option SUNW.SinstNM  code 11 = text;
option SUNW.SinstPTH code 12 = text;
option SUNW.SsysidCF code 13 = text;
option SUNW.SjumpsCF code 14 = text;
option SUNW.Sterm    code 15 = text;

#
# define SUNW jumpstart vendor options
#
class "solaris-sun4u" {
   match option vendor-class-identifier;
```

```
   vendor-option-space SUNW;
   next-server 1.2.3.4;
   option SUNW.SrootIP4 1.2.3.4;
   option SUNW.SrootNM  "js";
   option SUNW.SrootPTH "/opt/opsware/boot/jumpstart/Boot";
   option SUNW.SinstIP4 1.2.3.4;
   option SUNW.SinstNM  "js";
   option SUNW.SjumpsCF "js:/opt/opsware/boot/jumpstart/Conf";
   option SUNW.SsysidCF "js:/opt/opsware/boot/jumpstart/Conf";
   option SUNW.Sterm    "vt100";
   option SUNW.SbootFIL "/platform/sun4u/kernel/sparcv9/unix";
#
# We use a bogus install path just to give the installer
# something to mount for now.
#
   option SUNW.SinstPTH "/opt/opsware/boot/jumpstart/Boot";
   option dhcp-parameter-request-list 1,3,6,12,15,43,186,187;
}
#
# Begin dhcptool added SUNW client classes (do not edit)
#
subclass "solaris-sun4u" "FJSV.GPUU";
subclass "solaris-sun4u" "NATE.s-Note_737S";
subclass "solaris-sun4u" "NATE.s-Note_747S";
subclass "solaris-sun4u" "NATE.s-Note_777S";
subclass "solaris-sun4u" "SUNW.Netra-T12";
subclass "solaris-sun4u" "SUNW.Netra-T4";
subclass "solaris-sun4u" "SUNW.Sun-Blade-100";
subclass "solaris-sun4u" "SUNW.Sun-Blade-1000";
subclass "solaris-sun4u" "SUNW.Sun-Fire-15000";
subclass "solaris-sun4u" "SUNW.Sun-Fire-280R";
subclass "solaris-sun4u" "SUNW.Sun-Fire-480R";
subclass "solaris-sun4u" "SUNW.Sun-Fire-880";
subclass "solaris-sun4u" "SUNW.Sun-Fire";
subclass "solaris-sun4u" "SUNW.Ultra-1-Engine";
subclass "solaris-sun4u" "SUNW.Ultra-1";
subclass "solaris-sun4u" "SUNW.Ultra-2";
subclass "solaris-sun4u" "SUNW.Ultra-250";
subclass "solaris-sun4u" "SUNW.Ultra-30";
subclass "solaris-sun4u" "SUNW.Ultra-4";
subclass "solaris-sun4u" "SUNW.Ultra-5_10";
subclass "solaris-sun4u" "SUNW.Ultra-60";
subclass "solaris-sun4u" "SUNW.Ultra-80";
subclass "solaris-sun4u" "SUNW.Ultra-Enterprise-10000";
subclass "solaris-sun4u" "SUNW.Ultra-Enterprise";
subclass "solaris-sun4u" "SUNW.UltraAX-MP";
subclass "solaris-sun4u" "SUNW.UltraAX-e";
```



```
subclass "solaris-sun4u" "SUNW.UltraAX-e2";
subclass "solaris-sun4u" "SUNW.UltraAX-i2";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIe-NetraCT-40";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIe-NetraCT-60";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIi-Engine";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIi-Netract";
subclass "solaris-sun4u" "SUNW.UltraSPARC-IIi-cEngine";
subclass "solaris-sun4u" "SUNW.UltraSPARCengine_CP-20";
subclass "solaris-sun4u" "SUNW.UltraSPARCengine_CP-40";
subclass "solaris-sun4u" "SUNW.UltraSPARCengine_CP-60";
subclass "solaris-sun4u" "SUNW.UltraSPARCengine_CP-80";
#
# End dhcptool added SUNW client classes (do not edit)
#
# declare PXE vendor options
#
option space PXE;
option PXE.mtftp-ip             code 1  = ip-address;
option PXE.mtftp-cport          code 2  = unsigned integer 16;
option PXE.mtftp-sport          code 3  = unsigned integer 16;
option PXE.mtftp-tmout          code 4  = unsigned integer 8;
option PXE.mtftp-delay          code 5  = unsigned integer 8;
option PXE.discovery-control    code 6  = unsigned integer 8;
option PXE.discovery-mcast-addr code 7  = ip-address;
option PXE.boot-item            code 71 = unsigned integer 16;

#
# define PXE vendor options
#
class "pxeclients" {
   match if substring (option vendor-class-identifier, 0, 9) =
"PXEClient";
   vendor-option-space PXE;
   filename "pxelinux.0";
   next-server 1.2.3.4;
   option vendor-class-identifier "PXEClient";
#
# We set the MCAST IP address to 0.0.0.0 to tell the boot ROM we
# can't provide multicast TFTP, so it will have to use just
# plain ol' TFTP instead (address 0.0.0.0 is considered
# as "no address").
#
    option PXE.mtftp-ip 0.0.0.0;
    option dhcp-parameter-request-list = concat(dhcp-parameter-
    request-list,ba,bb);
}
```

Configuring the MS Windows DHCP Server for OS Provisioning

You may use the MS Windows DHCP server instead of the Opsware DHCP server to provision Windows or Linux on PXE 2.0 clients. The MS Windows DHCP server cannot be used during the OS provisioning of the following types of systems:

- Solaris
- PXE 0.99, 1.x clients (These older PXE clients have old PROMS and a PXE bootstrap floppy made with `rbfg.exe`.)

To configure the MS Windows DHCP server for OS Provisioning, perform the following steps:

1. On the MS Windows system running the DHCP server, you need to add option #60, so that it appears in the DHCP scope options. Open a command prompt, and enter the following command:

   ```
   netsh.exe dhcp server add optiondef 60 "PXEClient" STRING
   ```

2. Using the DHCP management snap-in (`dhcpmgmt.msc`), create a scope, which is usually a subnet declaration. In the scope options, #60 should now appear. Check the box, and then add the string `PXEClient`.

3. Using the same scope options box, configure options 66 and 67: Click the DHCP option #66 (Boot Server Host Name), and add the full DNS name of the tftp/boot server (for example `core01.test.com`). For option #67 (Bootfile Name), add the boot file name: `pxelinux.0`.

4. Ensure that the DHCP scope for the systems to be provisioned is set up with the required details, such as the DNS server, netmask, default router, DNS domain, and so forth.

5. At the command prompt, enter the following commands to locate the IP address of the Opsware Agent Gateway and the port forward for the Build Manager:

   ```
   netsh.exe dhcp server add optiondef 186 "buildmgr_ip" IPADDRESS
   netsh.exe dhcp server add optiondef 187 "buildmgr_port" WORD
   ```

6. Using the DHCP management snap-in (`dhcpmgmt.msc`), configure the options 186 and 187 to be part of your scope, and give them the appropriate values (IP address of the Opsware Agent Gateway and the port forward for the Build Manager, normally 8017).

7 Define option 043 (Vendor specific options) as a BINARY type, with the value `01 04 00 00 00 00 ff`. This setting tells the DHCP server to go directly to the ftp server specified in the Boot Server Host Name parameter, and also tells it to not use Multicast TFTP.

8 Restart the MS Windows DHCP server.

Configuring the Opsware and MS Windows DHCP Servers for OS Provisioning

You can configure the Opsware DHCP server to respond only to the OS provisioning requests (that is, from the PXE and Solaris clients), while the MS Windows DHCP server responds to all other requests.

1 Add the network subnet to the Opsware DHCP server. See "Configuring the Opsware DHCP Server for OS Provisioning" on page 147.

2 Stop the Opsware DHCP server with the following command:

```
/etc/init.d/opsware-sas stop dhcpd
```

3 Make a copy of the Opsware DHCP configuration file with the following commands:

```
cd /etc/opt/opsware/dhcpd
cp dhcpd.conf dhcpd.conf.orig
```

4 In a text editor, open the Opsware DHCP configuration file.

5 In the text editor, find the subnet definition you want to configure and comment out (with the # character) with the following lines:

```
range <IP1> <IP2>;
```

6 Immediately after the commented out line (# range), enter lines such as:

```
pool {
        allow members of "solaris-sun4u";
        allow members of "solaris-sun4us";
        allow members of "pxeclients";
        range <IP1> <IP2>;
        }
```

The preceding `pool` statement tells the DHCP server to continue serving the range specified, but only for the three types of clients indicated. (The first two `allow` statements are for Sun machines, the third is for PXE clients). In the preceding `pool` statement, be sure to include the closing brace }.

7. Repeat the preceding two steps for every subnet you wish to configure.
8. In the text editor, save the `dhcpd.conf` file.
9. Start the Opsware DHCP server:

   ```
   /etc/init.d/opsware-sas start dhcpd
   ```

10. Check the logs for DHCP errors. The DHCP service logs with `syslog`. See the `syslog.conf` file to determine how logging has been configured for the Opsware DHCP server.
11. Make sure that the MS Windows DHCP server subnet/scope declarations are changed to include the build manager DHCP options (code 186 and 187). See "Configuring the MS Windows DHCP Server for OS Provisioning" on page 154.
12. Make sure that the MS Windows DHCP server does not include options 43, 60, 66, or 67 in the scope/subnets you are configuring. This will prevent the PXE and Sun jumpstart clients from talking to the MS Windows DHCP server. Instead, they will talk to the Opsware DHCP server.
13. Make sure that the IP ranges of the MS Windows and Opsware DHCP servers don't overlap. As a guideline, the number of IP addresses in a given range should be twice the maximum number of servers that will be provisioned concurrently.
14. If the DHCP servers aren't directly connected to the network/subnet of the systems being provisioned, the DHCP requests must be forwarded to both DHCP servers, with the Opsware DHCP server being first.

Additional Network Requirements for OS Provisioning

OS Provisioning for Solaris

If you are using OS provisioning for Solaris (JumpStart) on an isolated network, you must have a default gateway (router) available, even if it does not route packets. For Solaris JumpStart to function properly, the IP address of the default gateway must be sent to the installation client that is being provisioned with DHCP. When you use the Opsware DHCP Configuration Tool, a default gateway is properly configured for Solaris because the DHCP Configuration Tool adds the default router appropriately.

Host Name Resolution

For Windows OS provisioning, the host name `buildmgr` should resolve on Windows installation clients.

The Opsware core host names must resolve using the DNS search order and DNS server information that the DHCP server provides. The DHCP server provides the DNS server IP address and the DNS search order. For each subnet you configure with the Opsware DHCP Tool, the DNS domain used by that subnet must have a DNS entry for `buildmgr`.

For example, you could have two subnets with the following domain names:

```
subnet1.example.com
subnet2.example.com.
```

Therefore, there must be two DNS entries:

```
buildmgr.subnet1.example.com
buildmgr.subnet2.example.com.
```

The host running the OS Provisioning Media Server must be able to resolve the IP address to the host name (a reverse lookup) of a server being provisioned.

See also "Host and Service Name Resolution Requirements" on page 72.

Open Ports

The server on which the OS is to be provisioned has the same requirements for connectivity to the Opsware core network as a managed server. See "Open Ports" on page 70.

Windows Patch Management Tasks

This section includes post-installation tasks for the Windows Patch Management feature of Opsware SAS.

Windows Patch Import Script

Before Windows patches can be installed on managed servers with Opsware SAS, the patches must be imported into the Software Repository. You can import the patches with the SAS Client or with the following shell script:

```
/opt/opsware/mm_wordbot/util/populate-opsware-update-library
```

This script downloads the Microsoft Patch Database and patches from the Microsoft site and imports them into the Software Repository. You should schedule the script to run weekly as a `cron` job on the Software Repository server. To end users of the SAS Client, the patches imported with the script appear to have been automatically imported.

To find out more about the script, see the "Automatically Importing Windows Patches" section in the *Opsware® SAS User's Guide: Application Automation*.

Patch Management on Windows NT 4.0 and Windows 2000

To use the `mbsacli.exe` patch utility for patch management on Windows NT 4.0 and Windows 2000, you must first install Internet Explorer 6.0 or later because the `mbsacli.exe` patch utility depends on it. This prerequisite is not required for Windows 2003 because IE 6.0 is pre-installed for this operating system.

To create a silent-installable version of IE 6.0 or later, use the Internet Explorer Administrator's Kit (IEAK) for the version of IE that you want to install. For more information on IEAK, see the following URL:

`http://microsoft.com/windows/ieak/default.asp`

To create a silent installable version of IE 6.0 or later, perform the following steps:

1. Install IEAK on your desktop system.
2. After you install IEAK, start the Internet Explorer Customization Wizard.
3. When creating the package, IEAK prompts for a Media Selection option. Select the option Flat (all files in one directory).
4. Select the defaults for all other options when you use the wizard.
5. After the wizard is complete, zip the contents of the directory it created. This directory contains the silent-installable version of IE.
6. To upload the ZIP package into Opsware SAS. See the *Opsware® SAS Policy Setter's Guide* for the steps to import software by using the SAS Client.
7. Set the following properties for the package when you import it into Opsware SAS. See the *Opsware® SAS Policy Setter's Guide* for the steps to edit the properties for a package in the SAS Client.
   - In the Installation Parameters section in the Install Flags field, enter the installation location:

     `%SystemDrive%\IE-redist`
   - In the Installation Parameters section in the Reboot Required field, select the Yes option.
   - In the Install Scripts section in the Post-Install Script tab, enter this text:

     `%SystemDrive%\IE-redist\ie6setup.exe /q:a /r:n`

     Where `ie6setup.exe` is the IE 6.x stub installer.

The /q:a install option specifies quiet install mode, with no user prompts. The /r:n install option suppresses restarting the server after IE installation.

8. Create a policy in the Software Policies and add the package to the policy. See the *Opsware® SAS Policy Setter's Guide* for the steps to create a software policy and add a package to a software policy.

9. Use SAS Client to install the necessary software on a Windows NT 4.0 or a Windows 2000 managed server. See the *Opsware® SAS User's Guide: Application Automation* for the steps to install software on a server by remediating a software policy onto a managed server.

Support for Redhat Network Errata and Channels

Red Hat Network is a system provided by the Red Hat company that enables system administrators to install and upgrade packages (RPMs) on Red Hat Linux servers. Included with Opsware SAS, the `rhn_import` CLI program enables you to download packages from Red Hat Network, upload the packages into Opsware SAS, and create software policies that correspond to Red Hat Network errata and channels. When you remediate the software policies, the packages in the policies are installed or upgraded on the managed servers.

You can import these packages and create software policies with the SAS Client, or you can perform these operations with `rhn_import`. To end users of the SAS Client, the actions performed by `rhn_import` appear to have occurred automatically. For more information on `rhn_import`, see "Automatically Importing Red Hat Network Errata" in the *Opsware® SAS Policy Setter's Guide*.

Opsware Global File System Tasks

This section contains optional post-installation tasks for the Opsware Global File System (OGFS).

Adding Instances of the OGFS to a Core

You can install multiple instances of the OGFS when you install an Opsware core. To do so, during the Opsware Installer interview, specify the IP addresses of the servers on which you plan to install the OGFS and follow the steps for installing an Opsware SAS component.

To install additional instances of the OGFS to an existing core, you must edit the following files:

- On the NFS server storing the user's `home` and `tmp` directories for the OGFS (the `ogfs.store.host` parameter in the response file), edit `/etc/exports` (on Linux) or `/etc/dfs/dfstab` (on Solaris) or add the IP address of the new OGFS server to allow it to mount the `ogfs.store.path` and `ogfs.audit.path` directories.
- On the NFS server storing the audit streams for the OGFS (the `ogfs.audit.host` parameter in the response file), edit `/etc/exports` (on Linux) or `/etc/dfs/dfstab` (on Solaris) or add the IP address of the new OGFS server to allow it to mount the `ogfs.store.path` and `ogfs.audit.path` directories.

  (The default value for both parameters is `theword`)

  See Chapter 5, "Opsware Global File System Prompts" on page 113 of this guide for more information.
- On the server that's running the Opsware Gateway, edit the `/etc/opt/opsware/opswgw-cgw0-<dcname>/opswgw.properties` file to add the ingress map for the new OGFS server; for example, add the following line:

  ```
  opswgw.IngressMap=<IP address of the new OGFS host>:HUB
  ```
- On each server that's running an OCC core component, edit the `/etc/opt/opsware/opswgw-lb/opswgw.properties` file by appending:

  ```
  :<IP address of the new OGFS host>:2222
  ```

  To the line:

  ```
  opswgw.LoadBalanceRule
  ```


Configuring User ID Numbers for the OGFS Server

When you install an Opsware SAS core, you can set values to control the range of UID and GID numbers used by the Opsware Global File System server. These values are used to provide unique user IDs for all Opsware users that are logged in to the OGFS server. When the Twist creates a new user, it will use these values to determine the next available (unique) user ID that is within the range for the local data center.

To set values that control the range of UID and GID numbers, you must specify the following Twist parameters in the `params.conf` file:

- **twist.min_uid**: Contains the minimum UID number that can be used. The default value is 80001.
- **twist.default_gid**: Contains the group ID number that a user is assigned to restrict Opsware users from using certain ports. The default value is 70001.

These parameters are specified as global in the `params.conf` file, which means that they will be written out to the global response file (`oiresponse.global`). This file is generated when the Model Repository export is performed on the source core server. When you follow the installation instructions and provide the global response file (`oiresponse.global`) as the initial response file to the destination core server, Opsware Installer will use the specified values.

For more information, see Table 5-7, “Opsware Global File System Prompts,” on page 113.

After you make changes to these parameters, you must restart the Twist server.

Chapter 8: Multimaster Installation

IN THIS CHAPTER

This section discusses the following topics:

- Multimaster Installation Basics
- Components of Multimaster Installations
- Converting a Standalone Core to Multimaster
- Adding a Core to a Multimaster Mesh
- Multimaster Post-Installation Tasks

This chapter describes how to run the Opsware Installer to upgrade a standalone core to a multimaster mesh and install target facilities. After providing a general outline of multimaster installation and its components, it details the process for converting a standalone core to a multimaster mesh. These instructions are followed by a short list of post-installation tasks.

Multimaster Installation Basics

An Opsware multimaster mesh contains two or more cores that communicate with each other. The first core installed in a multimaster mesh is the source core. The target core is the second, third, or subsequent core that you install in a multimaster mesh.

The following three steps represent the main phases in creating a multimaster mesh of cores:

1. Install a standalone (source) core.
   - Run the Opsware Installer interview, saving the data you enter at the prompts in a response file.
   - Run the installer again, specifying the response file, on one or more servers to install the Opsware components.
   - See “Standalone Core Installation” on page 128.
2. Convert the standalone core to a multimaster core.
   - Run the Opsware Installer interview with the response file created in the previous step, and then save your answers for this interview in another response file.
   - Run the installer again, specifying the latest response file, on one or more servers to add the multimaster components to the source core.
   - See “Converting a Standalone Core to Multimaster” on page 167.
3. Add the new target core to the multimaster mesh.
   - On the source core, run the Opsware Installer interview with the response file created in the previous step, and then save your answers for this interview in another response file.
   - Run the installer again, specifying the latest response file, and instruct the installer to define a new facility.
   - Run the installer again to export data from the Model Repository and to create a global response file.
   - Copy the export data file and the global response file from the source core server to the target core server.
   - On the target core, run the Opsware Installer interview with the global response file and save your answers for this interview in another response file.

- Run the installer again, specifying the latest response file, on one or more servers to install the components of the target core.
- See "Adding a Core to a Multimaster Mesh" on page 170.

For a given multimaster mesh, you perform steps 1 and 2 one time only. You perform step 3 every time you want to add another core to the multimaster mesh.

Components of Multimaster Installations

This section discusses prerequisites for installation and preexisting conditions that might effect your multimaster installation. It includes the following topics:

- Pre-Existing Core Installations
- Opsware Command Center (OCC)
- Multimaster Installation Prerequisites
- TIBCO Rendezvous

Pre-Existing Core Installations

If you installed a standalone core at any secondary facilities and you want to include these facilities in your multimaster mesh, you must perform the following tasks:

- Uninstall the Opsware core at the secondary facilities. See "Opsware Core Uninstallation" on page 209 in Chapter 11 for more information.
- Follow the instructions in the section "Multimaster Installation Basics" on page 164.

Opsware Command Center (OCC)

Target facilities (cores) in the multimaster mesh are not required to have an OCC component installed. Instead, you can manage the facility from any site in the multimaster mesh that does have an OCC installed. You need to install the OCC only if you want to manage your multimaster mesh locally from that facility or if you want to have a backup OCC.

TIBCO Rendezvous

In a multimaster mesh, Opsware SAS uses the TIBCO Certified Messaging system to synchronize Model Repositories at different facilities.

When you add a core to a multimaster mesh, the Opsware Installer automatically configures the TIBCO Rendezvous routing daemon (`rvrd`). For more information, see "TIBCO Rendezvous Configuration for Multimaster" on page 249.

Multimaster Installation Prerequisites

Perform the following tasks in preparation for installing a multimaster core:

- Plan your Opsware System deployment. When planning for a core, you must decide whether you want to install the core components on a single server or on multiple servers. See Chapter 1, "Opsware SAS Architecture" and "Opsware Core Performance Scalability" on page 49.
- Perform the pre-installation administration tasks, such as configuring the network. See Chapter 3, "Pre-Installation Requirements."
- Gather information in preparation for the Opsware Installer interview. This information includes the name and ID of the facility for the core. See Chapter 5, "Prerequisites for the Installer Interview."
- Verify that every Opsware core server has a unique IP address within the entire multimaster mesh.
- To support a multimaster conversion, verify that the required state of every Opsware core server is fully running.
- After you synchronize the time on all servers within a facility, synchronize the time between the facilities in the multimaster mesh. Synchronize the time with an external time-server that uses Network Time Protocol (NTP) so that all servers are using the same Coordinated Universal Time (UTC).
- Verify that the multimaster installation meets the same network requirements as a standalone installation, except that each core must be on a different Local Area Network (LAN or VLAN). The cores must be in different broadcast domains.
- Ensure that each core in a mesh has a different subdomain so that managed servers can resolve the unqualified host names `spin`, `way`, and `theword`.
- Verify that the `tnsnames.ora` file on the source core contains entries for every Model Repository in the mesh. For example entries, see "tnsnames.ora: Multimaster Mesh Requirements" on page 233. If the `tnsnames.ora` file of the source core does not contain entries for the target cores, the Multimaster Tools page in the Opsware Command Center will not work.

- Ensure that you do not have conflicting Oracle software versions within the multimaster mesh. See “Multiple Oracle Versions and Multimaster Cores” on page 218.

Converting a Standalone Core to Multimaster

This section describes how to convert a standalone core into a multimaster mesh. The core to be converted is referred to as the source core. (If you already have a multimaster mesh and want to add an additional core, go to “Adding a Core to a Multimaster Mesh” on page 170.)

To convert a core from standalone to multimaster, perform the following steps:

1. Obtain the Opsware SAS installation media for this release.

   See “Installation Media for the Opsware Installer” on page 116, including the recommendation, “Copying the DVD to a Local Disk.”

2. On each server of the source core, mount the Product Software DVD or NFS-mount the directory that contains a copy of the DVD contents.

   The Opsware Installer must have read/write root access to the directories where it installs Opsware components, even NFS-mounted network appliances.

3. On the Model Repository server in the source core, log in as root.

4. Change to the root directory:
   ```
   cd /
   ```

5. Start the Opsware Installer with the `-r` (response file) and the `--interview` options. For example:

   ```
   /opsware_system/opsware_installer/install_opsware.sh -r
   /usr/tmp/oiresponse.stand_single --interview
   ```

   You must specify the full path to the script. The directory path in the preceding command indicates that you copied the Opsware SAS Product Software DVD to a local disk or network share using the required directory structure.

   You should run the Opsware Installer with the response file that you created when you installed the source core. If this response file is not available, invoke the Opsware Installer with no command line options, and the interview will automatically start.

   The Opsware Installer displays the following options:

   ```
   Welcome to the Opsware Installer. Please select one of the
   following installation options:
   ```

```
1 - Standalone Installation: Standalone Opsware Core
2 - Multimaster Installation: First Core (convert from
standalone)
3 - Multimaster Installation: Define New Facility; Export
Model Repository
4 - Multimaster Installation: Additional Core
```

6 At the installation options prompt, select the following option:

```
2 - Multimaster Installation: First Core (convert from
standalone)
```

7 At the interview mode prompt, select one of the following options:

```
1 - Simple Interview Mode
2 - Advanced Interview Mode
```

Option 1 is for using default values for many of the configuration parameters. Option 2 is for specifying all configuration parameters during the interview.

8 Respond to the interview prompts.

The installer displays default values in square brackets [ ].

See “The Opsware Installer Interview Mode” on page 92.

9 Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

10 Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.stand_to_mm]
```


The response file is a text file that contains the answers you entered during the interview. You can enter the name of the response file or accept the default. In either case, write down the name of the response file. Note that the default file name corresponds to the type of installation.

11 The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file. Select one of the following options:

- If you are satisfied with the responses you entered in the interview and you are ready to install the Model Repository Multimaster Additions now, enter `y` to continue.
- If you do not want to install the Model Repository Multimaster Additions now, enter `n`.

12 If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, invoke the Opsware Installer with the `-r` option to specify the response file created by the latest interview. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r
/usr/tmp/oiresponse.stand_to_mm
```

13 At the components prompt, select one or more components to install:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Model Repository (truth), Multimaster Additions
2 ( ) Data Access Engine (spin), Multimaster Component
3 ( ) Multimaster Infrastructure Components (vault)
4 ( ) Command Engine (way), Multimaster Component
5 ( ) Software Repository (word), Multimaster Component
6 ( ) Opsware Global Filesystem Server, Multimaster Component
7 ( ) Opsware Command Center (OCC), Multimaster Component
Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection:
```

You must install the components in the order they are listed. For example, you must install the Model Repository Multimaster Additions first.

If you are installing all of the components on a single server, then you can enter `a` for all. If you do not select `a`, then you must run the Opsware Installer again (as shown in the preceding step) and select the remaining components.

14 If you are installing the components on multiple servers, follow the instructions in this step. (If you are installing the components on a single server, skip this step.)

Copy the response file generated by the installer interview to all other servers in the source core.

On each server in the source core, run the Opsware Installer with the `-r` option, as shown in step 12. Select and install the remaining components from the menu shown in step 13.

You must install each multimaster addition on the same server running the corresponding standalone component. For example, install the Model Repository Multimaster Additions on the server running the standalone Model Repository, and install the Data Access Engine Multimaster Component on the server running the standalone Data Access Engine. Although not required, the Model Repository Multimaster Component (vault) is usually installed on the same server as the Model Repository.

15 Follow the instructions in the section “Adding a Core to a Multimaster Mesh” on page 170.

Adding a Core to a Multimaster Mesh

This section describes in detail how to add a new Opsware core to a multimaster mesh. There are several cross references, so ideally, you should scan this section first and make sure that you are prepared to perform all of the steps.

Throughout this section, the first core in the mesh is referred to as the source core. The new core that you are adding is called the target core. (If you do not have a multimaster mesh, you are reading the wrong section; go to the section “Converting a Standalone Core to Multimaster” on page 167.)

In an Opsware SAS core, the Opsware Model Repository uses an Oracle database. This section provides the instructions for installing an Opsware SAS core with Oracle 10g by using the Opsware Installer.

When you use an existing Oracle database, you must configure the Oracle database instance correctly to work with the Opsware SAS core. For information about installing an Opsware SAS core by using an existing Oracle database, see Appendix A, “Oracle Setup for the Model Repository” in this guide.

Before proceeding with the installation, follow the instructions in “Multimaster Installation Prerequisites” on page 166.

To add a new core to a multimaster mesh, perform the following steps:

1. Obtain the Opsware SAS installation media for this release.

   See “Installation Media for the Opsware Installer” on page 116, including the recommendation, “Copying the DVD to a Local Disk.”

2. On the Model Repository server of the source core and on each server of the target core, mount the Product Software DVD or NFS-mount the directory that contains a copy of the DVD contents.

   The Opsware Installer must have read/write root access to the directories where it installs Opsware components, even NFS-mounted network appliances.

3. On the Model Repository server in the source core, invoke the Opsware Installer with the `-r` (response file) and the `--interview` options. For example:

   ```
   /opsware_system/opsware_installer/install_opsware.sh -r
   /usr/tmp/oiresponse.stand_to_mm --interview
   ```

   You must specify the response file created when you converted the core from standalone to multimaster.

   The Opsware Installer displays the following options:

   ```
   Welcome to the Opsware Installer. Please select one of the
   following installation options:

   1 - Standalone Installation: Standalone Opsware Core
   2 - Multimaster Installation: First Core (convert from
   standalone)
   3 - Multimaster Installation: Define New Facility; Export
   Model Repository
   4 - Multimaster Installation: Additional Core
   ```

4. At the installation options prompt, select the following option:

   ```
   3 - Multimaster Installation: Define New Facility; Export
   Model Repository
   ```

5. At the interview mode prompt, select one of the following options:

   ```
   1 - Simple Interview Mode
   2 - Advanced Interview Mode
   ```

Option 1 is for using default values for many of the configuration parameters. Option 2 is for specifying all configuration parameters during the interview.

6 Respond to the interview prompts.

The installer displays default values in square brackets [ ].

For the short name of the target core (`slaveTruth.dcNm` parameter), enter a new facility name. This name must be unique within the multimaster mesh.

See "The Opsware Installer Interview Mode" on page 92.

7 Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

8 Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.add_dc_to_mesh]
```

The response file is a text file that contains the answers you entered during the interview. You can enter the name of the response file or accept the default. In either case, write down the name of the response file. Note that the default file name corresponds to the type of installation.

9 The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file. Select one of the following options:

- If the Opsware Gateway in the source core is on a different server than the Model Repository, enter `n`. Copy the response file to the server with the Opsware Gateway and go on to the next step.
- If you are satisfied with the responses you entered in the interview and you are ready to define the new facility now, enter `y` to continue.

- If you do not want to define the new facility now, enter `n`.

10 If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, log in to the server running the Opsware Gateway and invoke the installer with the `-r` option. Be sure to specify the response file created by the latest interview. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r
/usr/tmp/oiresponse.add_dc_to_mesh
```

11 At the components prompt, select the following option:

```
1 ( ) Define New Facility
```

Wait for the installer to finish this operation before going on to the next step. The Opsware Installer enters the target facility in the Model Repository of the source core, automatically generating the target facility's ID.

12 Find the ID of the target facility.

To find the facility ID, perform the following steps:

- Log in to the SAS Web Client as the `admin` user at the source facility.
- From the navigation panel, click Facilities under Environment.
- Click the link for the target facility. Write down the facility ID.

In step 13 through step 21, you perform the tasks for exporting data from the Model Repository of the source core.

If you are adding a third (or more) core to a multimaster mesh, you can export data from a core other than the original source core. In this case, the instructions are slightly different, as noted in step 15 on page 174 and step 38 on page 179.

13 On the servers where the Opsware Command Center (OCC) and the Opsware Global File System (OGFS or hub) are installed, stop the Web Services Data Access Engine (twist) by entering the following command:

```
/etc/init.d/opsware-sas stop twist
```

14 On the servers where the Data Access Engine (spin) is installed, stop the engine by entering the following command:

```
/etc/init.d/opsware-sas stop spin
```

If the OCC and the Data Access Engine are installed on different servers, you must also run the preceding command on the OCC server.

15 On the server running the Model Repository Multimaster Component, wait for all transactions to be published by examining the `/var/log/opsware/vault/log` file.

If the log contains successive entries "QUERIED THE DATABASE" and does not contain recent "SENDING TRANSACTION" entries, the transactions from the installation have been published.

If you are going to export data from a core other than the original source core, wait for the transactions to propagate to the core that will be exported before performing step 18 on page 174.

16 On the server where the Model Repository Multimaster Component (vault) is installed, stop the engine by entering the following command:

```
/etc/init.d/opsware-sas stop vaultdaemon
```

17 Log in to the server running the Model Repository and invoke the installer with the `-r` option to specify the response file created by the latest interview. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r
/usr/tmp/oiresponse.add_dc_to_mesh
```

18 At the components prompt, select the following option:

```
2 ( ) Export Model Repository (truth)
```

The installer exports the data from the Model Repository into the `truth_data.tar.gz` file, which by default resides in the directory `/var/opt/opsware/truth`. (You specified this directory at the `truth.dest` prompt of the interview.)

Depending on the amount of data, the export might take 20 minutes or more. To track the progress of the export in a different window, run the following command.

```
tail -f /var/log/opsware/install_opsware/truth
/truth_exp<number>.log
```

Before you export the data from the Model Repository so that it can be imported into the target core, make sure that you do not have conflicting Oracle software versions within the multimaster mesh. See "Multiple Oracle Versions and Multimaster Cores" on page 218.

19 On the source core servers where the Data Access Engine (spin) is installed, start the engine by entering the following command:


```
/etc/init.d/opsware-sas start spin
```

If the OCC and the Data Access Engine are installed on different servers, you must also run the preceding command on the OCC server.

20 On the servers where the OCC and the Opsware Global File System Server (OGFS or hub) are installed, start the Web Services Data Access Engine (twist) by entering the following command:

```
/etc/init.d/opsware-sas start twist
```

21 On the server where the Model Repository Multimaster Component (vault) is installed, start the engine by entering the following command:

```
/etc/init.d/opsware-sas start vaultdaemon
```

Examine the logs for the Model Repository Multimaster Component to ensure that it started properly. These logs are located in the following directory:

```
/var/log/opsware/vault
```

The log files are named `log`, `log.1`, `log.2`, `log.3`, and so forth.

22 Copy the Model Repository export file (`truth_data.tar.gz`) to the server where you will install the Model Repository in the target core.

The Unix `oracle` user needs read access to the `truth_data.tar.gz` file on the Model Repository host in the target core.

23 Copy the global response file (`oiresponse.global`) from the source core server of the Model Repository to the target core server on which you will install the new Model Repository.

On the source core, the `oiresponse.global` file resides in the same directory as the Model Repository export file. The default directory is `/var/opt/opsware/truth`.

24 On the target core servers, make the following directory:

```
mkdir -p /var/opt/opsware/crypto/cadb/realm
```

25 Copy the database of cryptographic material and the Unix Tar file Gzipped from the source core server (that is running the Model Repository) to every target core server. The database of cryptographic material and the Unix Tar file Gzipped are in the following files:

```
/var/opt/opsware/crypto/cadb/realm/opsware-crypto.db.e
```

```
/var/opt/opsware/crypto/cadb/realm/opsware-crypto.tgz.e
```

The full path name of the file on the target core servers must match the preceding lines. The root user requires read access to the directory and files.

26 Log in to the target core server on which you will install the Model Repository and invoke the Opsware Installer. Specify the `-r oiresponse.global` file and the `--interview` options. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r
/usr/tmp/oiresponse.global --interview
```

Be sure to specify the global response file that you copied to the target core.

The Opsware Installer displays following options:

```
Welcome to the Opsware Installer. Please select one of the
following installation options:

1 - Standalone Installation: Standalone Opsware Core
2 - Multimaster Installation: First Core (convert from
standalone)
3 - Multimaster Installation: Define New Facility; Export
Model Repository
4 - Multimaster Installation: Additional Core
```

27 At the installation options prompt, select the following option:

```
4 - Multimaster Installation: Additional Core
```

28 At the interview mode prompt, select one of the following options:

```
1 - Simple Interview Mode
2 - Advanced Interview Mode
```

Option 1 is for using default values for many of the configuration parameters. Option 2 is for specifying all configuration parameters during the interview.

29 At the database configuration option prompt, select the following option:

```
1 - Install Oracle with Opsware
```

For information about installing an Opsware SAS core by using option 2 ("`Use Existing Oracle Database`"), see Appendix A, "Oracle Setup for the Model Repository". When you use an existing Oracle database, you must configure the Oracle database instance correctly to work with the Opsware SAS core.

30 Respond to the interview prompts.

The installer displays default values in square brackets [ ]. Unless you have changed the source core, do not change the values that were in the global response file you copied from the source core. Note the following requirements for the prompts:

- The facility ID, short name, and subdomain must match the values generated when the target facility was defined in the source core. You wrote down the facility ID in step 12 on page 173.
- The authorization domain must match the value provided for the source core.
- The path to the data export file, `truth_data.tar.gz`, in the target core must match the path you used when copying the file from the source core.
- The path for the OS provisioning media must already exist on the server where you will install the OS Provisioning Media Server component.

31 Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

32 Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.mm_subs]
```

The response file is a text file that contains the answers you entered during the interview. You can enter the name of the response file or accept the default. In either case, write down the name of the response file. Note that the default file name corresponds to the type of installation.

33 The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file. Select one of the following options:

- If you are satisfied with the responses you entered in the interview and you are ready to install the Model Repository now, enter `y` to continue.

- If you do not want to install the Model Repository now, enter `n`.

34 If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, invoke the Opsware Installer with the `-r` option to specify the response file created by the interview. For example:

```
/opsware_system/opsware_installer/install_opsware.sh -r
/usr/tmp/oiresponse.mm_subs
```

35 At the components prompt, select one or more components to install:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Oracle RDBMS for SAS
2 ( ) Model Repository (truth), Secondary Core
3 ( ) Data Access Engine (spin), Multimaster Component
4 ( ) Multimaster Infrastructure Components (vault)
5 ( ) Command Engine (way), Multimaster Component
6 ( ) Software Repository (word), Multimaster Component
7 ( ) Opsware Global Filesystem, Multimaster Component
8 ( ) Opsware Global Filesystem Server (OGFS)
9 ( ) Opsware Command Center (OCC), Multimaster Component
10 ( ) OS Provisioning Media Server
11 ( ) OS Provisioning Build Manager
12 ( ) Opsware Gateway, Secondary Core
13 ( ) OS Provisioning Boot Server
Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection:
```

You must install the components in the order they are listed. For example, you must install the Model Repository first.

If you are installing all of the components on a single server, then you may enter `a` for all. If you do not select `a`, then you must run the Opsware Installer again (as shown in the preceding step) and select the remaining components. (If you are installing the components on multiple servers, see the next step.)

Installing the Opsware Global File System Server (OGFS) component is supported on servers running Sun Solaris 9 and Red Hat Enterprise Linux 3 AS.

36 If you are installing the components on multiple servers, follow the instructions in this step. (If you are installing the components on a single server, skip this step.)

Copy the response file generated by the installer interview to all other servers in this core.

Copy the `tnsnames.ora` file from the server with the Model Repository to the other core servers. Make sure that the path for the file (`/var/opt/oracle/tnsnames.ora`) is the same on all core servers. See "tnsnames.ora File Requirements" on page 233.

On each server in this core, run the Opsware Installer with the `-r` option, as shown in step 34. Select and install the remaining components from the menu shown in step 35.

If the Model Repository exists on a server with no other Opsware components installed on it, you must install an Opsware Agent on that server. See the *Opsware® SAS User's Guide: Server Automation* for instructions.

37 (Optional) If you are distributing the core components across multiple servers, you can install additional instances of the following components:

- Data Access Engine

  If you install more than one Data Access Engine, then you must perform the procedure described in "Reassigning the Data Access Engine to a Secondary Role" in the *Opsware® SAS Administration Guide*.

- OS Provisioning Media Server
- Opsware Command Center (OCC)
- Opsware Global File System (OGFS)

  To install multiple instances of the OGFS when you install an Opsware core, during the Opsware Installer interview, specify the IP addresses of the servers on which you plan to install the OGFS.

  To install additional instances of the OGFS to an existing core, you must perform manual steps. See Chapter 7, "Adding Instances of the OGFS to a Core" on page 160 of this guide for more information.

38 If you exported data from a core other than the original source core, you might need to configure TIBCO manually.

By default, the target core will try to connect to the original source core. If you want the target core to connect to a different core then you must configure TIBCO manually and edit the Opsware Gateway properties file. For instructions, see "Adding a TIBCO Rendezvous Neighbor" on page 251.

39 Perform the tasks in Chapter 7, "Standalone Core Post-Installation Tasks" on page 135 of this guide.

40 Perform the post-installation tasks.

Multimaster Post-Installation Tasks

After you add a new core to a multimaster mesh, perform the tasks described in this section.

Associating Customers with a New Facility

Associate the appropriate customers with each new facility so that servers managed at that facility are associated with the correct customers accounts. For more information, see the Customer Account Administration section of the *Opsware® SAS Policy Setter's Guide*.

Updating Permissions for New Facilities

After you add new facilities to your multimaster mesh, your Opsware users will not have the required permissions to access these new facilities. To grant access, you must assign the required permissions to the user groups. For more information, see the User Group and Setup section of the *Opsware® SAS Administration Guide*.

Verifying Multimaster Transaction Traffic

To verify multimaster transaction traffic with the target facility, perform the following steps:

1 Log in to the SAS Web Client as a user that belongs to the Opsware System Administrators group.

2 From the navigation panel, click Multimaster Tools under Administration. The View window appears.

3 In the State View Window, note the color of the status box beside each transaction.

A transaction is a unit of change to a Model Repository database that consists of one or more updates to rows and has a globally unique transaction ID. If the transactions with the target facility are green, the new Opsware core is integrated into the multimaster mesh. It is normal for some of the transactions to have an orange status (not sent) for a while.

4 Click **Refresh** to refresh the cached data.

For more information, see the Opsware Multimaster Mesh Administration section in the *Opsware® SAS Administration Guide*.

Chapter 9: Satellite Installation

IN THIS CHAPTER

This section discusses the following topics:

- Satellite Installation Basics
- Satellite Installation Requirements
- Gateway Configuration for a Satellite
- Satellite Installation
- Post-Satellite Installation Tasks

This chapter provides an overview of satellite installation and satellite installation requirements. The majority of this chapter explains how to install a satellite, including detailed steps and post installation tasks.

Satellite Installation Basics

An Opsware Satellite manages servers in a remote data center. The following steps provide an overview of the Satellite installation process. For detailed instructions, see “Satellite Installation” on page 194.

1. Obtain the Opsware SAS installation DVDs.
2. Run the Opsware Installer (`install_opsware.sh` script) in interview mode. The interviewer prompts you for information about your environment and saves the information in a response file.
3. Run the Opsware Installer and select the Opsware Gateway from the list of components to install. The Opsware Installer launches the Opsware Gateway Installer.
4. Respond to the Opsware Gateway Installer prompts.
5. Run the Opsware Installer (`install_opsware.sh` script) and select the other components to install.

Satellite Installation Requirements

Before you install an Opsware Satellite, verify that the following requirements are met.

Required Open Ports

The ports listed in Table 9-1 must be open for the Opsware Gateway in a Satellite. The ports in the table are the default values. (You may select other values during the installation.)

Table 9-1: Open Ports for a Satellite Gateway

PORT	PROPERTY NAME IN OPSWARE GATEWAY PROPERTIES FILE	DESCRIPTION
2001	`opswgw.TunnelDst`	The port for a tunnel end-point listener. This port will be used if you install other Gateways that tunnel to the Gateway on this Satellite.
3001	`opswgw.ProxyPort`	The proxy port on which the Agents contact the Gateway.

Table 9-1: Open Ports for a Satellite Gateway

PORT	PROPERTY NAME IN OPSWARE GATEWAY PROPERTIES FILE	DESCRIPTION
4040	`opswgw.IdentPort`	The port of the Gateway's `ident` service, which is used by the Software Repository Cache.

If you are going to install the OS Provisioning Boot Server and Media Server in the Satellite, then additional ports must be open. For a list of these ports, see Table 3-8 on page 71.

Required Entries in /etc/hosts

The Software Repository Cache of the Satellite requires the following entries in the `/etc/hosts` file:

```
127.0.0.1  theword
127.0.0.1  wordcache
```

Other Requirements for a Satellite Installation

- The Satellite server must meet the requirements listed in "Supported Operating Systems: Opsware Core Server" on page 45.

The supported operating systems for the OS Provisioning components are more restricted than those for the other Satellite components (Gateway and Software Repository Cache). See the *Opsware Policy Setter's Guide* for more information.

- The Satellite server must have the necessary packages listed in "Solaris and Linux Requirements" on page 56.
- The Opsware core for this Satellite is up and running.
- The Satellite server must have network connectivity to the server running the core Gateway.
- In the SAS Web Client for the core, you must be able to log in as a member of the Administrators group (`admin`) and as a member of a group that has the Manage Gateway permission.

- You must have root access on the core server so that you can copy the database of cryptographic material from the core to the Satellite server.
- The Satellite server uses UTC, as described in "Time and Locale Requirements" on page 77. The time of the Satellite server must be synchronized with the core server.
- When using network storage for the Software Repository Cache, the network storage configuration must allow root write access over NFS to the directories where the Software Repository Cache is to be installed.
- If you are going to install the OS Provisioning Boot Server and Media Server in the Satellite, then see the requirements in "DHCP Proxying" on page 73.
- You must know how to edit files with the `vi` editor. The Opsware Gateway Installer launches the `vi` editor, which you will use to edit a properties file.

Required Packages for SuSE Linux Enterprise Server 9

In addition to the packages listed in "Solaris and Linux Requirements" on page 56, a Satellite on this version of Linux requires the `compat-2004.7.1-1` package.

Gateway Configuration for a Satellite

This section illustrates various Satellite topologies and the corresponding settings in the Gateway properties files. In the diagrams, the arrows between Gateways represent tunnels. (A tunnel is a TCP connection between two Gateways that carries multiplexed TCP or UDP connections.) The boxes labelled with the letter "A" designate managed servers, which run Opsware Agents.

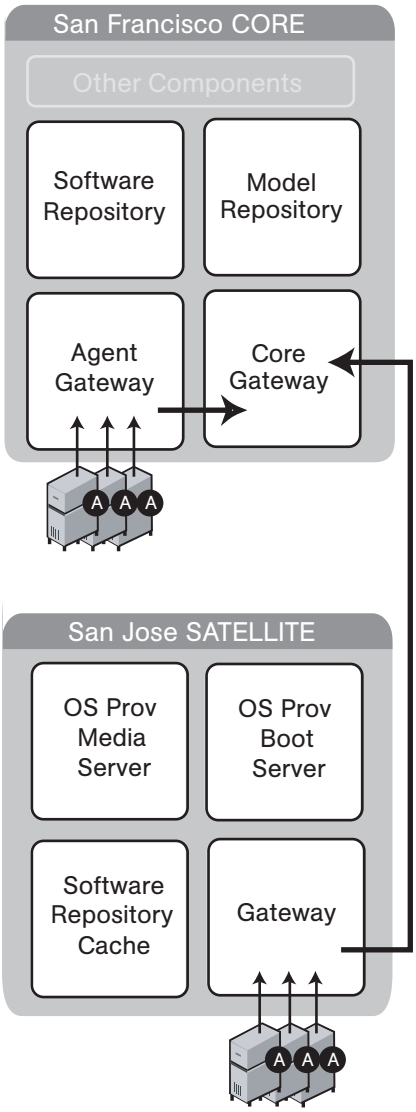
Satellite with a Standalone Core

Figure 9-1 shows a single Opsware Satellite that has a tunnel to a standalone core. In this example, the main facility is in San Francisco, and a smaller remote facility is in San Jose.

The core is made up of several components, including the Software Repository, the Model Repository, and two gateways. The figure does not show other required core components, such as the Command Engine, but indicates them with an ellipsis (...) button. When you install a standalone core, the Opsware Installer creates both the Agent and core Gateways. The Agents in the San Francisco facility communicate with the core through the Agent Gateway. The Agents in the San Jose facility connect to the San Francisco core via TCP connections to the Satellite Gateway.

In a Satellite, the Software Repository Cache and Gateway components are required. The Software Repository Cache contains local copies of software packages to be installed on managed servers in the Satellite. The Gateway multiplexes connections into and out of the Satellite via one or more tunnels. Optionally, a Satellite can contain the OS Provisioning Boot Server and Media Server components.

Figure 9-1: Single Satellite with a Standalone Core



The following listing shows a few entries in the Gateway properties file of the San Jose Satellite.

In the properties file, the `opswgw.GWAddress` specifies the IP address or host name where the Satellite Gateway runs. When a new Gateway is added to a realm, the value of the `opswgw.GWAddress` is dynamically added to the list of Gateways that Agents in the realm can communicate with. (A realm is a routable group of IP addresses.) The Agent installer and the `opswgw.GWAddress` must both specify either IP addresses or host names. For example, if the Agent installer specifies an IP address in its `opsw_gw_addr_list` option, then the `opswgw.GWAddress` must also specify an IP address, not a host name. If host names are used, they must be resolvable (with DNS or `/etc/hosts`) by the Agents that contact this Gateway. Specifying IP addresses is recommended because it is less error prone. (This document shows host names in the example diagrams and listings because they are easier to read.)

The `opswgw.Realm` specifies the realm of the Gateway. A realm is a logical name for a group of IP addresses that can be contacted by a particular set of Gateways. Realms enable Opsware SAS to manage servers with overlapping IP addresses. (This situation can occur when the servers in a remote facility are behind NAT devices or firewalls.) The realm plus the IP address uniquely identifies a managed server. Servers with overlapping IP addresses must reside in separate realms.

The `opswgw.TunnelSrc` has five parameters. The first two parameters identify the remote host (`sanfran.myops.com`) and port (2001) where the core Gateway listens for connections. Note that the host and port of the `opswgw.TunnelSrc` in the Satellite must match those of the `opswgw.TunnelDst` in the core. The next two parameters of `opswgw.TunnelSrc` specify the cost and bandwidth of the tunnel. (See “Configuring Routing (Cost)” on page 190 and “Limiting Bandwidth” on page 194.) The last parameter (`.../opswgw.pem`) is a certificate file in the Privacy Enhanced Mail (PEM) format. If you specify the certificate file, the data transmitted through the tunnel will be encrypted using SSL. The header of the certificate file includes the cipher choice and authentication options.

The `opswgw.DoNotRouteService` and `opswgw.HijackService` properties are required for this Satellite Gateway because the Satellite includes a Software Repository Cache. With these properties, if an Agent has a request for the Software Repository, the Satellite Gateway routes the request to the local Software Repository Cache.

The `opswgw.ProxyPort` identifies the port on the Satellite through which the Agents contact the Gateway. The `opswgw.IdentPort` is for an identity service used by the Software Repository Cache.

Typically, you'll use the default ports for the properties. However, you must enter the hosts for the `opswgw.GWAddress` and `opswgw.TunnelSrc` properties. The following listing shows some of the entries in the Gateway properties file for the San Jose Satellite. (Although the `opswgw.TunnelSrc` entry wraps around to the next line in this listing, in the actual properties file, the entry is on a single line.)

```
opswgw.Gateway=SanJose
opswgw.Realm=SanJose
opswgw.GWAddress=sanjose.myops.com
opswgw.TunnelSrc=sanfran.myops.com:2001:10:0:/var/opt/opsware/
crypto/SanJose/opswgw.pem
opswgw.DoNotRouteService=theword:1003
opswgw.DoNotRouteService=127.0.0.1:1003
opswgw.HijackService=wordcache:1003
opswgw.ProxyPort=3001
opswgw.IdentPort=4040
```

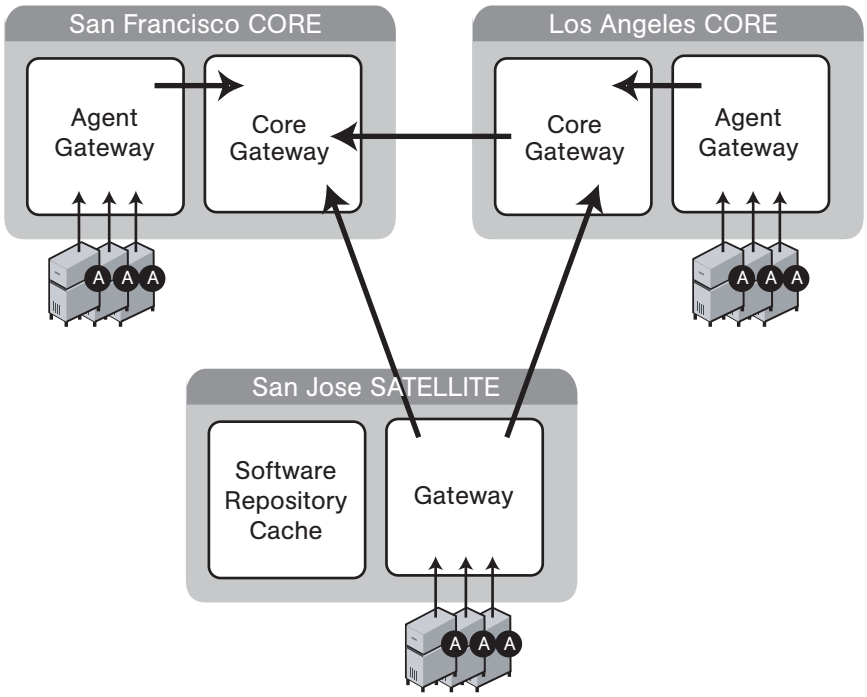
The following lines are from the core Gateway properties file of the San Francisco facility:

```
opswgw.Gateway=cgw0-SanFrancisco
opswgw.Realm=SanFrancisco
opswgw.TunnelDst=2001:/var/opt/opsware/crypto/cgw0-
SanFrancisco/opswgw.pem
```

Satellite in a Multimaster Mesh

Figure 9-2 shows two cores, San Francisco and Los Angeles, in a multimaster mesh. The multimaster traffic passes through the core Gateways. The Gateway in the San Jose Satellite points to both core Gateways. In this example, the communication link between the San Jose and San Francisco facilities is the fastest and has the most bandwidth. During normal operations, the servers in San Jose are managed by the San Francisco core. If the connection between San Jose and San Francisco fails, then the Gateway in San Jose will communicate instead with the core in Los Angeles. (See "Configuring Routing (Cost)" on page 190.)

Figure 9-2: Single Satellite in a Multimaster Mesh



The lines that follow are from the properties file of the Satellite Gateway in San Jose. The first `opswgw.TunnelSrc` property points to the San Francisco Gateway; the second one points to the Los Angeles Gateway. Both lines indicate that the core Gateways use the default port (2001) to listen for connections.

```
opswgw.Gateway=SanJose
opswgw.Realm=SanJose
opswgw.TunnelSrc=sanfran.myops.com:2001:100:0:/var/opt/opsware/
crypto/SanJose/opswgw.pem
opswgw.TunnelSrc=losang.myops.com:2001:200:0:/var/opt/opsware/
crypto/SanJose/opswgw.pem
```

Configuring Routing (Cost)

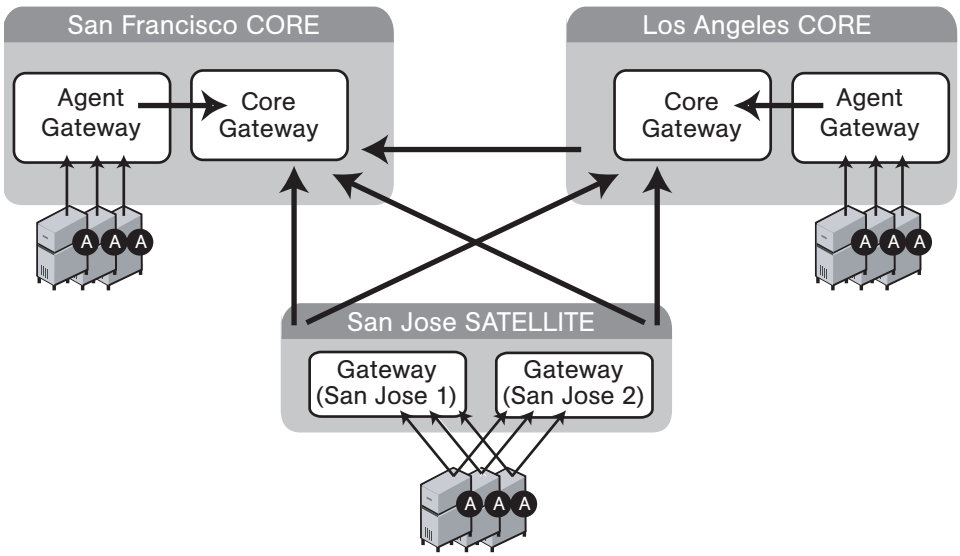
A Satellite Gateway routes traffic to only one core Gateway at any given time. The Gateway chooses the route with the lowest cost. The cost is the third parameter of the `opswgw.TunnelSrc` property. In the preceding listing, the `opswgw.TunnelSrc` properties specify that the cost from San Jose to San Francisco is 100 and the cost

between San Jose and Los Angeles is 200. Therefore, the Satellite Gateway will use the connection to San Francisco, unless for some reason that connection becomes unavailable.

Multiple Gateways in a Satellite

The topology shown in Figure 9-3 provides failover capability in two ways. First, each Gateway in the San Jose Satellite tunnels to both core Gateways. If one core becomes unavailable, the other core can manage the servers in the Satellite. Second, the Agents in the San Jose Satellite point to both Satellite Gateways. If one Satellite Gateway becomes unavailable, the Agents on the managed servers can communicate with a core Gateway via the other Satellite Gateway. Both Gateways in San Jose must belong to the same realm. An Agent can communicate with any Gateway in the same realm.

Figure 9-3: Multiple Gateways in a Satellite



The following lines are from the core Gateway properties file of the San Francisco facility:

```
opswgw.Gateway=cgw0-SanFrancisco
opswgw.Realm=SanFrancisco
opswgw.TunnelDst=2001:/var/opt/opsware/crypto/cgw0-
SanFrancisco/opswgw.pem
```

The core Gateway properties file of the Los Angeles facility has similar entries:

```
opswgw.Gateway=cgw0-LosAngeles
opswgw.Realm=LosAngeles
```

```
opswgw.TunnelDst=2001:/var/opt/opsware/crypto/cgw0-LosAngeles/
opswgw.pem
opswgw.TunnelSrc=sanfran.myops.com:2001:1:0:/var/opt/opsware/
crypto/cgw0-LosAngeles/opswgw.pem
```

The lines that follow are from the properties file of the first Gateway in the San Jose Satellite:

```
opswgw.Gateway=SanJose1
opswgw.Realm=SanJose
opswgw.TunnelSrc=sanfran.myops.com:2001:100:0:/var/opt/opsware/
crypto/SanJose1/opswgw.pem
opswgw.TunnelSrc=losang.myops.com:2001:200:0:/var/opt/opsware/
crypto/SanJose1/opswgw.pem
```

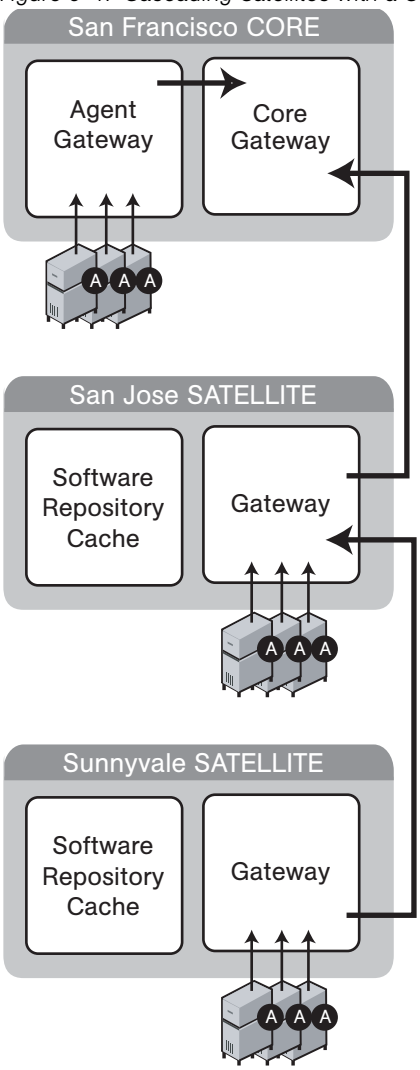
The next lines are from the properties file of the second Gateway in the San Jose Satellite:

```
opswgw.Gateway=SanJose2
opswgw.Realm=SanJose
opswgw.TunnelSrc=sanfran.myops.com:2001:100:0:/var/opt/opsware/
crypto/SanJose2/opswgw.pem
opswgw.TunnelSrc=losang.myops.com:2001:200:0:/var/opt/opsware/
crypto/SanJose2/opswgw.pem
```


Cascading Satellites

Figure 9-4 is an example of cascading Satellites, a topology in which Satellite Gateways are connected in a chain. These Satellite Gateways must be in different realms. (For more information, see "Managing the Software Repository Cache" in the *Opsware® SAS Administration Guide*.)

Figure 9-4: Cascading Satellites with a Standalone Core



The following lines are from the core Gateway properties file of the San Francisco facility:

```
opswgw.Gateway=cgw0-SanFrancisco
opswgw.Realm=SanFrancisco
```

```
opswgw.TunnelDst=2001:/var/opt/opsware/crypto/cgw0-
SanFrancisco/opswgw.pem
```

The lines that follow are from the Gateway properties file of the San Jose Satellite.

```
opswgw.Gateway=SanJose
opswgw.Realm=SanJose
opswgw.TunnelDst=2001:/var/opt/opsware/crypto/SanJose/
opswgw.pem
opswgw.TunnelSrc=sanfran.myops.com:2001:100:0:/var/opt/opsware/
crypto/SanJose/opswgw.pem
```

The next lines are from the Gateway properties file of the Sunnyvale Satellite:

```
opswgw.Gateway=Sunnyvale
opswgw.Realm=Sunnyvale
opswgw.TunnelSrc=sanjose.myops.com:2001:100:256:/var/opt/
opsware/crypto/Sunnyvale/opswgw.pem
```

Limiting Bandwidth

In Figure 9-4, suppose that the tunnel between Sunnyvale and San Jose shares a 512 kilobit/sec DSL connection with another application. Since this connection is relatively slow, you might want to limit the tunnel bandwidth to 256 kilobits/sec. To limit the bandwidth, you specify 256 for the fourth parameter of the `opswgw.TunnelSrc` property. (See the previous listing of the Sunnyvale properties file.) If you do not want to limit the tunnel bandwidth, set this parameter to 0. Note that the bandwidth parameter is not used to determine the cost of a route. (See “Configuring Routing (Cost)” on page 190.)

Satellite Installation

This section describes how to create a new Opsware Satellite with the simple topology shown in Figure 9-1. This topology has the following characteristics:

- The Satellite contains one Opsware Gateway and one Software Repository Cache, installed on the same server.
- The Satellite Gateway communicates with one core Gateway. No other Gateways communicate with the Satellite Gateway.

Required Information

You will be prompted for the following information during the installation process:

- The password to decrypt cryptographic material. During the installation of the core, the Opsware Installer prompts for this password.
- The IP address of the server running the core Gateway.
- The IP address of the server on which you will install the Satellite Gateway.
- The port of the tunnel destination of the core Gateway. (The default port is 2001.) The core Gateway listens on this port for a connection from the Satellite Gateway. In the core Gateway properties file, this port is the value of the `opswgw.TunnelDst` property. On the core Gateway server, the path of the properties file is as follows:

  `/etc/opt/opsware/opswgw-cgw0-<facility>/opswgw.properties`
- The Opsware user name (`admin`) and password of a user that belongs to the Administrators group.
- The name of the new Gateway in the Satellite. The new Gateway will be installed in the following directory:

  `/opt/opsware/opswgw/bin`
- The name of the new realm to be serviced by the Gateway in the Satellite. Opsware SAS uses the realm name and the IP address of a managed server to uniquely identify a managed server. The Opsware Gateway Installer assigns the realm name to the new facility name of the Satellite. The core and Satellite facility names will be different. You may want to name the realm according to the physical location of the Satellite's data center, for example, the building, corporate site, or city. The SAS Web Client lists the facility names of the core and its Satellites.

Installing a Satellite

If an Opsware SAS Agent is already installed in the Satellite, it must be uninstalled before running the Satellite installer. You must deactivate or delete the Satellite server from Opsware SAS so that you do not have an existing Agent on a server on which you are going to install it.

As a result of the installation process, the Satellite server is owned by the "Opsware" customer. Any effects on access rights should be taken into account.

This section contains the step-by-step instructions for running the Opsware Installer (`install_opsware.sh` script).

1. Obtain the Opsware Server Automation System (SAS) installation media.

See “Installation Media for the Opsware Installer” on page 116, including the recommendation, “Copying the DVD to a Local Disk.”

2. On the server where you will install the new Opsware Satellite, mount the Satellite Base DVD or the Satellite Base Including OS Provisioning DVD or NFS-mount the directory that contains a copy of the DVD contents.

Whether you choose to install the “Satellite Base” DVD or the “Satellite Base Including OS Provisioning” DVD depends on whether you install the OS Provisioning components in the satellite. See “Installation Media for the Opsware Installer” on page 116 for information about each of the Opsware SAS DVDs.

The Opsware Installer must have read/write root access to the directories where it installs Opsware components, even NFS-mounted network appliances.

3. In a terminal window, log in as root.
4. Make the following directory:

   ```
   mkdir -p /var/opt/opsware/crypto/cadb/realm
   ```

5. Copy the database of cryptographic material and the Gzipped tar file from the core server to the Satellite server. On the core server, this database and the Gzipped tar file are in the following files:

   ```
   /var/opt/opsware/crypto/cadb/realm/opsware-crypto.db.e
   ```

   ```
   /var/opt/opsware/crypto/cadb/realm/opsware-crypto.tgz.e
   ```

   The database of cryptographic material and the Gzipped tar file must be copied to the same directory and file names on the Satellite server. The directory and database and the Gzipped tar file need to be readable by the root user.

6. Change to the root directory:
   ```
   cd /
   ```
7. Run the Opsware Installer in interview mode by invoking it with no command-line options:
   ```
   /opsware_system/opsware_installer/install_opsware.sh
   ```


You must specify the full path to the script. The directory path shown in this step indicates that you copied an Opsware SAS Satellite DVD (the Satellite Base DVD or the Satellite Base Including OS Provisioning DVD) to a local disk or a network share using the required directory structure.

8 At the interview mode prompt, select one of the following options:

```
1 - Simple Interview Mode
2 - Advanced Interview Mode
```

Option 1 is for using default values for many of the configuration parameters. Option 2 is for specifying all configuration parameters during the interview.

9 Respond to the interview prompts.

The `cgw_address prompt` is for the core Gateway, not the Satellite Gateway. For more information on the prompts, see Table 5-6 on page 112.

10 Decide if you want to finish the interview.

When you enter all of the required information, the Opsware Installer displays this message:

```
All parameters have values. Do you wish to finish the
interview (y/n):
```

If you are satisfied with your answers, press y.

If you want to review or change your answers, press n. The installer displays the prompts again, showing in brackets [ ] the values that you previously entered.

11 Create the response file.

When you are finished with the interview, the installer prompts you for the name of the response file:

```
Name of response file to write
[/usr/tmp/oiresponse.satellite]
```

The response file is a text file that contains the answers you entered during the interview. You can enter the full path and name of the response file or accept the default. Note that the default file name corresponds to the type of installation.

12 The Opsware Installer prompts you to indicate whether you want to continue the installation by using the response file:

```
Would you like to continue the installation using this
response file? (y/n):
```

If you are satisfied with the responses you entered in the interview and you are ready to install the Satellite now, enter `y` to continue. If you do not want to install the Satellite now, enter `n`.

13 If you entered `y` in the previous step, skip this step. If you entered `n` in the previous step, invoke the Opsware Installer with the `-r` option to specify the response file created by the interview:

```
/opsware_system/opsware_installer/install_opsware.sh -r
<full_path_to_response_file>
```

14 At the components prompt, select 1 to install the Opsware Gateway. The components prompt follows:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Opsware Gateway (Interactive Install)
2 ( ) Software Repository Cache (wordcache)
3 ( ) OS Provisioning Boot Server
4 ( ) OS Provisioning Media Server
Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection: 1
```

Note that you must install the components in the order they are listed.

The selections for the OS Provisioning Boot Server and OS Provisioning Media Server only appear if you are running the installation from the Satellite Base Including OS Provisioning DVD.

15 Verify that the Opsware Installer launches the Opsware Gateway Installer, which displays the following banner:

```
************************************************************
*                                                          *
*                Opsware Gateway Installer                 *
*         Copyright (C) 2004-2006: Opsware Inc.            *
*                   support@opsware.com                    *
*                                                          *
************************************************************
```

16 Verify that you have the necessary information for the Gateway, as described in "Required Information" on page 194. The Opsware Gateway Installer displays the following message:

```
For a new install please have the following information
available before you begin:
```



```
1) Opsware administrator username and password.
2) The Realm name this Gateway will service.
3) If the Realm is new what type will it be.
4) The unique Gateway name for this Gateway.

Are you ready to proceed? [y/n]
```

17 At the proceeding prompt, enter y. The Opsware Gateway Installer displays the following lines:

```
================================================
ISM install
================================================
. . .
```

18 Enter the name of the realm for the Opsware Gateway you are installing. The prompt for the realm follows:

```
================================================
Create/Verify Realm
================================================
Enter the Gateway's Realm name:
You entered '<realm-name>', is this correct [y/n]
```

19 There are three ways for the installer to contact the Opsware core. At the prompt for the option number, enter 3. The installer displays the following lines:

```
I must now contact an Opsware Core to continue the
intallation...
There are three ways this can be done:
 1) Via an existing Gateway's ProxyPort
 2) Via direct connections (no NATs)
 3) Via a temporary (local) Gateway
Enter option number: 3
```

20 Enter the IP address of the server running the core Gateway at the following prompt:

```
Enter IP of a remote GW:
```

21 Enter the tunnel destination port of the core Gateway at the following prompt. The default port is 2001. (For more information, see “Required Information” on page 194.)

```
Enter TunnelDst port of the remote GW: 2001
```

22 At the following prompt, enter y.

```
Is the tunnel listener at <ip-addr:port>
using SSL? [y/n] y
```

23 Enter the user name (`admin`) and password of an Opsware user that belongs to the Administrators group. The user name and password prompts follow.

```
================================================
```

```
Connect to Opsware
================================================

Log in to Opsware as an administrator

Enter username:admin
Enter password:
```

24 Verify that the Opsware Gateway Installer displays the following lines:

```
================================================
Checking time synchronization
================================================

Gateway time looks good.
```

25 At the prompt that follows, enter 1 to create a new Satellite.

```
================================================
Configure Realm
================================================

The realm '<realm-name>' does not exist.  You have two
options:
  1) Create a new Satellite DC named '<realm-name>'.
  2) Add a new Realm, '<realm-name>', to an existing DC.
  3) Exit.
Enter option number: 1
```

26 At the following prompt, enter the name for the new Opsware Gateway that you are installing.

```
================================================
Gateway Configuration
================================================

Enter the Gateway's name:
```

27 Verify that the Opsware Gateway Installer opens the properties file in the `vi` text editor. The following lines are at the top of the properties file:

```
#####################################################
#
# Opsware Gateway Properties file for a SAT Gateway
#
#####################################################
```

The full path name of the properties file follows:

`/etc/opt/opsware/opswgw-cgw0-<facility>/opswgw.properties`

28 For the `opswgw.GWAddress` property, enter the IP address of the host on which you are installing this Gateway (that is, the host you are logged in to now). Example:

```
opswgw.GWAddress=192.168.198.92
```

29 For the `opswgw.TunnelSrc` property, change the placeholder IP address of 10.0.0.11 to the IP address of the host running the core Gateway. The port following the IP address is the tunnel destination of the core Gateway. (The default port is 2001.) Example:

```
opswgw.TunnelSrc=192.168.165.242:2001:100:0:/var/opt/
opsware/crypto/<gateway-name>/opswgw.pem
```

30 Because you are going to install a Software Repository Cache (wordcache) in a later step, verify that the following lines in the Opsware Gateway Properties file are not commented out:

```
opswgw.DoNotRouteService=theword:1003
opswgw.DoNotRouteService=127.0.0.1:1003
opswgw.HijackService=wordcache:1003
```

31 After you've finished editing the Opsware Gateway Properties in `vi`, save the file and exit `vi`.

32 Respond to the prompts that ask if you'd like to proceed. The Opsware Gateway Installer performs several more tasks and displays the following messages:

```
Gateway Crypto Generation
. . .
Wordcache Crypto Generation
. . .
Starting Opsware Gateway
. . .
Verify Gateway Startup
```

When it's finished, the Opsware Gateway Installer displays the following line:

```
Opsware Gateway Installed!
```

33 Invoke the Opsware Installer with the `-r` option to specify the response file created by the interview in step 11 on page 197:

```
/opsware_system/opsware_installer/install_opsware.sh -r
<full_path_to_response_file>
```

34 At the components prompt, select one or more components to install:

```
Welcome to the Opsware Installer.
Please select the components to install.
1 ( ) Software Repository Cache (wordcache)
2 ( ) OS Provisioning Boot Server
3 ( ) OS Provisioning Media Server
```

```
Enter a component number to toggle ('a' for all, 'n' for
none).
When ready, press 'c' to continue, or 'q' to quit.

Selection:
```

You must install the components in the order they are listed. For example, you must install the Software Repository Cache before the OS Provisioning Boot Server.

The Software Repository Cache is required and must be installed on the same server as the Gateway.

The selections for the OS Provisioning Boot Server and OS Provisioning Media Server only appear, if you are running the installation from the Satellite Base Including OS Provisioning DVD.

The OS Provisioning Boot Server and Media Server are required only if you want to use the Opsware OS Provisioning feature in the Satellite. The OS Provisioning Boot Server and Media Server can reside on a different server than the Gateway and Software Repository Cache. (See step 35.)

If you are installing all of the components on the same server, then you may enter `a` for all. If you do not select `a`, then you must run the Opsware Installer again (specifying the response file) and select the remaining components.

35 If you are installing the OS Provisioning components on a different server than the other Satellite components, follow the instructions in this step.

- Copy the database of cryptographic material and the Gzipped tar file from the server with the Satellite Gateway to the server that will run the OS Provisioning components. Here is the full path of to these files:

  `/var/opt/opsware/crypto/cadb/realm/opsware-crypto.db.e`

  `/var/opt/opsware/crypto/cadb/realm/opsware-crypto.tgz.e`

  The database of cryptographic material and the Gzipped tar file must be copied to the same directory. The directory and files need to be readable by the root user.

- Copy the response file generated by the installer interview to the server that will run the OS Provisioning components.

- On the server that will run the OS Provisioning components, run the Opsware Installer with the `-r` option, as shown in step 33. Select and install the remaining components from the menu shown in step 34.

Post-Satellite Installation Tasks

After you install a Satellite, perform the tasks listed in the following sections. For more information, see the Opsware Satellite Administration section of the *Opsware® SAS Administration Guide*.

Facility Permission Settings

The Opsware Gateway Installer assigns the realm name to the facility name of the Satellite. To access managed servers in the Satellite, an Opsware user must belong to a group that has the necessary permissions for the Satellite's facility. Until you set the facility permissions, Opsware users cannot view or modify the managed servers associated with the Satellite's facility. For example, you might set the permissions for the Satellite facility to Read & Write for the Advanced Users group, enabling members of this group to modify the servers managed by the Satellite.

For instructions, see "Setting the Facility Permissions of a User Group" in the *Opsware® SAS Administration Guide*.

Checking the Satellite Gateway

To verify that the core Gateway is communicating with the Satellite Gateway, perform the following steps:

1. Log in to the SAS Web Client as a member of a users group that has the Manage Gateway permission.
2. From the navigation panel, click Administration ➤ Gateway.
3. Verify that the upper left corner of the Manage Gateway page displays a link for the new Satellite Gateway.

   If the Manage Gateway page does not display the link for the Satellite, you might need to correct the properties file of the Satellite Gateway. The full path name of the properties file follows:

   `/etc/opt/opsware/opswgw-cgw0-<facility>/opswgw.properties`

   If you modify the properties file, you must restart the Satellite Gateway:

   `/etc/init.d/opsware-sas restart opswgw-cgw0`
4. Log in to the SAS Web Client as a member of a users group that has the Read (or Read & Write) permission on the Satellite's facility.
5. From the navigation panel, click Servers ➤ Manage Servers.
6. Verify that the Manage Server page displays the host name of the Satellite server.

Enabling the Display of Realm Information

By default, the SAS Web Client does not display realm information, which is needed by users who manage Gateways and Software Repository Caches.

To enable access to the realm information, perform the following steps:

1. Log into the SAS Web Client as a user that belongs the Administrators group and to a group that has the Configure Opsware permission.
2. From the navigation panel, click Administration ➤ System Configuration.
3. Select the Opsware Server Automation System Web Client link.
4. In the System Configuration page, for the name `owm.features.Realms.allow`, type the value `true`.
5. Click **Save**.

DHCP Configuration for OS Provisioning

After you install the OS Provisioning Boot Server component, you must set up a DHCP server. For more information, see “DHCP Configuration for OS Provisioning” on page 144.

Chapter 10: Opsware SAS Configuration

IN THIS CHAPTER

This section discusses the following topic:

- Opsware SAS Configuration
- Configure e-mail Alerts
- Set Up Opsware Groups and Users
- Set Up Software Repository Replicator
- Create Opsware Customers
- Define Policies for Software Management
- Install Opsware Agents on Existing Servers
- Prepare Opsware SAS for OS Provisioning
- Prepare Opsware SAS for Patch Management
- Establish Monitoring Practices for Opsware SAS

Opsware SAS Configuration

After you complete the tasks in the preceding sections of this guide, the core components of Opsware SAS should be running and you should be able to log in to the SAS Web Client. You can now configure Opsware SAS so that end users can start managing servers in their operational environment.

The following tasks provide a general outline of the configuration process:

Configure e-mail Alerts

The Opsware managed servers, the multimaster mesh, and the Opsware Code Deployment and Rollback feature can be configured to send e-mail alerts. Your e-mail administrator should set up the Opsware core and managed servers as sendmail clients. In the SAS Web Client, you should configure the e-mail alerts before you install

Agents on the managed servers. See the *Opsware® SAS Administration Guide* for information

Set Up Opsware Groups and Users

To log in to the SAS Web Client, you specify a user name and password. Each user belongs to a group, and each group has a set of permissions that control access to features (actions), managed servers, and folders. See the *Opsware® SAS Administration Guide* for information.

Set Up Software Repository Replicator

After you install an Opsware core in multimaster mode, you can set up replications for the Software Repository in a facility. See the *Opsware® SAS Administration Guide* for information.

Create Opsware Customers

When you ran the Opsware Installer for a standalone core, you specified a default customer. You may also create and assign new customers to the facility. See the *Opsware® SAS Policy Setter's Guide* for information.

Define Policies for Software Management

See the *Opsware® SAS Policy Setter's Guide* for information.

Install Opsware Agents on Existing Servers

After you install an Opsware Agent, the server may be managed with Opsware SAS. See the *Opsware® SAS User's Guide: Server Automation* for information.

Prepare Opsware SAS for OS Provisioning

When you provision (install) an OS on a server, Opsware SAS automatically installs an Agent. See the *Opsware® SAS Policy Setter's Guide* for information.

Prepare Opsware SAS for Patch Management

See the *Opsware® SAS User's Guide: Application Automation* for information.

Establish Monitoring Practices for Opsware SAS

Perform the following tasks:

- Run the Agent reachability tests in the SAS Web Client. See the *Opsware® SAS User's Guide: Server Automation* for information.
- Run the diagnostic tests in the SAS Web Client. See the *Opsware® SAS Administration Guide* for information.
- Review the Opsware SAS component log files. See the *Opsware® SAS Administration Guide* for information.

Chapter 11: Opsware Core Uninstallation

IN THIS CHAPTER

This section discusses the following topics:

- Uninstall Basics
- Procedures for Uninstalling Cores
- Uninstalling a Standalone Core
- Uninstalling One Core in a Multimaster Mesh
- Uninstalling All Cores in a Multimaster Mesh
- Decommissioning a Facility with the SAS Web Client

After a general overview of the uninstallation process, this chapter shows how to uninstall a standalone core, remove a core from a multimaster mesh, and uninstall an entire Opsware SAS made up of multiple cores in different facilities.

Uninstall Basics

There are several reasons that you might choose to uninstall an Opsware core. Before installing Opsware SAS in a production environment, you might want to uninstall the Opsware core after you finish testing it. Or, if you are consolidating facilities, you might want to uninstall an Opsware core in one facility in preparation for moving it to another facility.

Before you uninstall an Opsware core, Opsware Inc. recommends that you back up the Oracle database running on the server where the Model Repository is installed. See "Oracle Database Backup Methods" on page 243.

Uninstalling the Model Repository permanently deletes all data in the database. But when you uninstall an Opsware core, you can choose to preserve the Opsware SAS data in the Model Repository database. If you choose to preserve this data, the Opsware Installer stops the uninstallation.

Stopping the uninstallation gives you the opportunity to back up the data in the Model Repository. After you begin the Model Repository uninstallation, the Opsware Installer will not preserve any data in the Model Repository.

You can choose to preserve or remove all the packages stored on the Software Repository. You can also choose to preserve the database of cryptographic material for the Opsware core. If you choose to preserve crypto, the database of cryptographic material will be saved; otherwise it will be deleted when the uninstallation finishes.

Procedures for Uninstalling Cores

You can perform the following four uninstallation procedures according to your system needs:

- Uninstalling a Standalone Core
- Uninstalling One Core in a Multimaster Mesh
- Uninstalling All Cores in a Multimaster Mesh
- Decommissioning a Facility with the SAS Web Client

Uninstalling a Standalone Core

To uninstall a standalone core, perform the following steps:

1. Before you uninstall the Opsware core components from the servers running them, you should deactivate the servers with the SAS Web Client. Otherwise, if you try to re-install an Opsware core component on one of the servers later, the installation will fail. (For more information, see "Deactivating a Server" in the *Opsware® SAS User's Guide: Server Automation*.)
2. Log in as root.
3. Change to the root directory:
   ```
   cd /
   ```
4. Run the `uninstall_opsware.sh` script:
   ```
   /opsware_system/opsware_installer/uninstall_ opsware.sh -r
   <response-file>
   ```
5. At the components prompt, select one or more components to uninstall:
   ```
   Welcome to the Opsware Installer.
   Please select the components to uninstall.
   1 ( ) Opsware Gateway
   2 ( ) OS Provisioning Build Manager
   3 ( ) OS Provisioning Media Server
   4 ( ) OS Provisioning Boot Server
   5 ( ) Opsware Command Center (OCC)
   6 ( ) Opsware Global Filesystem Server (OGFS)
   7 ( ) Software Repository (word)
   8 ( ) Command Engine (way)
   9 ( ) Data Access Engine (spin)
   10 ( ) Model Repository (truth)
   11 ( ) Oracle RDBMS
   ```
   If the Opsware Gateway does not run on a separate server, uninstall it last.
6. Remove the `/var/opt/opsware/install_opsware` directory.

If you indicated at the prompt that you want to preserve crypto (the database of cryptographic material), you should *not* delete the `/var/opt/opsware/crypto` directory. Deleting this directory deletes the database of cryptographic material.

Uninstalling One Core in a Multimaster Mesh

When uninstalling a core from a multimaster mesh, you should not uninstall the source core unless you are planning to uninstall the entire mesh.

See "Uninstalling All Cores in a Multimaster Mesh" on page 213 in this chapter for more information.

To uninstall one core in a multimaster mesh, perform the following steps:

1. Log in to any SAS Web Client that is still online and perform the following tasks:
   1. Using the System Configuration feature, update the `listeners` configuration parameter by removing the entry for the core that is being uninstalled. Update the listeners parameter by selecting "Model Repository, Multimaster Component" in the System Configuration page.
   2. If a Data Access Engine that is being uninstalled is currently serving as the multimaster central role, a Data Access Engine in another core must be selected to serve as Multimaster Central.

      See "Reassigning the Data Access Engine to a Secondary Role" in the *Opsware® SAS Administration Guide*.
   3. Verify that all transactions have propagated to the other facilities, except for the facility that is being uninstalled.

      See "Verifying Multimaster Transaction Traffic" on page 180.
2. Decommission the facility for the core you are uninstalling. See "Decommissioning a Facility with the SAS Web Client" on page 214.
3. Restart the Model Repository Multimaster Component in all cores except the core that is being uninstalled by entering the following command as root on the server running the engine:

   ```
   /etc/init.d/opsware-sas stop vaultdaemon
   ```

   ```
   /etc/init.d/opsware-sas start vaultdaemon
   ```
4. Stop the OCC component in the core that is being uninstalled by entering the following command as root:

   ```
   /etc/init.d/opsware-sas stop occ.server
   ```
5. In the core that is being uninstalled, stop all Data Access Engines.

Log in as root to the server where the Data Access Engine is running and enter the following command:

```
/etc/init.d/opsware-sas stop spin
```

6. If the OCC and the Data Access Engine are installed on different servers, you must also run the `spin stop` command on the OCC server.

7. Stop the Model Repository Multimaster Component in the core that is being uninstalled by entering the following command as root on the server running the engine:

   ```
   /etc/init.d/opsware-sas stop vaultdaemon
   ```

8. Restart the Data Access Engine that is serving as Multimaster Central by entering the following commands as root:

   ```
   /etc/init.d/opsware-sas stop spin
   ```

   ```
   /etc/init.d/opsware-sas start spin
   ```

9. For the core that you are uninstalling, on each server running an Opsware component, run the following script.

   ```
   /opsware_system/opsware_installer/uninstall_opsware.sh
   ```

   Uninstall the components by following the instructions in step 4 through step 6 in the section "Uninstalling a Standalone Core."

Uninstalling All Cores in a Multimaster Mesh

To uninstall all cores in a multimaster mesh, perform the following steps:

1. Stop the OCC by logging on as root to the server where the OCC is running and enter the following command:

   ```
   /etc/init.d/opsware-sas stop occ.server
   ```

2. Stop the Data Access Engine.

   Log in as root to the server where the Data Access Engine is running and enter the following command:

   ```
   /etc/init.d/opsware-sas stop spin
   ```

   If the OCC and the Data Access Engine are installed on different servers, you must also run the `stop spin` command on the OCC server.

3. Stop the Model Repository Multimaster Component in all cores by logging i to the servers running the engines and entering the following command as root:

   ```
   /etc/init.d/opsware-sas stop vaultdaemon
   ```

4. In each core, uninstall the Opsware components on the servers where they are installed.

   ```
   /opsware_system/opsware_installer/uninstall_opsware.sh
   ```

   Follow the instructions in step 4 through step 6 in the section "Uninstalling a Standalone Core."

Decommissioning a Facility with the SAS Web Client

Performing this procedure does not shut down or uninstall Opsware SAS in a facility. Decommission facilities with care, because this task cannot be undone.

When you decommission a facility, the facility is still listed in the SAS Web Client, however, it is grayed out. After a short name is used, even if it is decommissioned, that name cannot be reused.

To decommission a facility, perform the following steps:

1. In the SAS Web Client, deactivate the server running the core of the facility that you wish to decommission. (For instructions, see "Deactivating a Server" in the *Opsware® SAS User's Guide: Server Automation*.)
2. From the navigation panel, click **Environment ➤ Facilities**. The Facilities page appears.
3. Select the facility that you want to decommission.
4. On the Properties tab, note the answer to the following question:

   ```
   Is this facility in use?
   ```

   If the answer is No, the **Decommission** button is displayed.
5. Click **Decommission**.

Appendix A: Oracle Setup for the Model Repository

IN THIS APPENDIX

This appendix discusses the following topics:

- Oracle RDBMS Install Basics
- Supported Oracle Versions
- Oracle RDBMS Hardware Requirements
- Required Operating System Packages and Patches
- Opsware-Installed Oracle vs. a Standard Oracle RDBMS
- Pre-Oracle Universal Installer Tasks
- Manually Creating the Oracle Database
- Post-Create the Oracle RDBMS Tasks
- Database Monitoring Strategy
- Troubleshooting System Diagnosis Errors
- Garbage Collection
- Oracle Database Backup Methods
- Useful SQL
- Model Repository Installation on a Remote Database Server

This appendix explains how to install, configure, and maintain an Oracle database to support the Opsware Model Repository.

Oracle RDBMS Install Basics

The Model Repository (truth) is an Opsware core component that stores information in an Oracle database.

Opsware SAS provides an Oracle version 10g database that you can use the Opsware Installer to create. You can also use an existing Oracle database, or an Oracle database you create using the Oracle Universal Installer. Note that, you will need to configure these databases to support the Opsware Model Repository.

The process for installing Oracle and the Model Repository has the following three major steps:

1. Install the Oracle RDBMS software.
2. Create the Oracle database (instance).
3. Install the Model Repository.

You can perform both Steps 1 and 2 by using the Opsware Installer or by using the Oracle Universal Installer. You can perform Step 3 only by using the Opsware Installer. Not that, the Oracle database must be created before you install the Model Repository, whether you use the Opsware Installer to install and create the database or use the Oracle Universal Installer.

Oracle RDBMS Install using the Opsware Installer

The Opsware Installer performs steps 1 and 2 as a single procedure, installing Oracle version 10g. If you intend to perform steps 1 and 2 using the Opsware Installer, see “Opsware-Installed Oracle vs. a Standard Oracle RDBMS” on page 222.

Oracle RDBMS Install using a Standard Oracle Installation

If you intend not to use the Opsware Installer but rather use the Oracle Universal Installer, or use an existing Oracle database, then you should read the following sections:

- “Pre-Oracle Universal Installer Tasks” on page 226
- “Manually Creating the Oracle Database” on page 228
- “Post-Create the Oracle RDBMS Tasks” on page 232

Supported Oracle Versions

Support for the Model Repository is limited to certain versions of Oracle running on certain versions of operating systems. Table A-1 lists the supported Oracle versions.

Table A-1: Supported Oracle Versions for Model Repository

ORACLE EDITION	VERSIONS
Oracle Standard Edition	9.2.0.4 9.2.0.6 9.2.0.7 9.2.0.8 10.2.0.2
Oracle Standard Edition One	10.2.0.2
Oracle Enterprise Edition	9.2.0.4 9.2.0.6 9.2.0.7 9.2.0.8 10.2.0.2

Oracle version 9.2.0.5 is not supported with Opsware SAS.

To be supported on the Model Repository, the Oracle versions listed in Table A-1 are limited to the operating systems listed in Table A-2.

Table A-2: Supported Operating Systems for Model Repository

SUPPORTED OPERATING SYSTEMS FOR MODEL REPOSITORY	VERSIONS	ARCHITECTURE
Sun Solaris	Solaris 8 Solaris 9 Solaris 10	Sun SPARC Sun SPARC Sun SPARC
Red Hat Linux	Red Hat Enterprise Linux 3 AS	32 bit x86

Table A-2: Supported Operating Systems for Model Repository

SUPPORTED OPERATING SYSTEMS FOR MODEL REPOSITORY	VERSIONS	ARCHITECTURE
Red Hat Linux	Red Hat Enterprise Linux 4 AS	64 bit x86

Multiple Oracle Versions and Multimaster Cores

For the database export to succeed during the installation of a multimaster core, the version of the target (slave) database cannot be 9.x if the version of the source (master) database is 10.x. Table A-3 lists these allowed version combinations.

Table A-3: Database Versions Allowed for Multimaster

SOURCE DB VERSION	TARGET DB VERSION	ALLOWED?
9	9	Y
9	10	Y
10	9	N
10	10	Y

Oracle RDBMS Hardware Requirements

The server that will run the Oracle database for the Model Repository has the following hardware requirements.

Physical Memory and Swap Space

Oracle requires at least 1024 MB of physical RAM. The amount of swap space required depends on the size of the physical RAM, as shown in Table A-4.

Table A-4: RAM and Swap Space

SIZE OF RAM (MB)	SWAP SPACE REQUIRED (MB)
1024 - 2048	1.5 times the size of RAM
2094 - 8192	equal to size of RAM
more than 8192	9

Temporary Disk Space

The Oracle Universal Installer (OUI) requires up to 400 MB free space in the `/tmp` directory.

Permanent Disk Space

The amount of disk space required depends on the Oracle edition and the number of servers managed by Opsware SAS, as listed in Table A-5.

Table A-5: Database Versions Allowed for Multimaster

ORACLE EDITION	DISK SPACE REQUIRED BY ORACLE RDBMS SOFTWARE (GB)	ADDITIONAL DISK SPACE (FOR DATA AND INDEX TABLESPACES) REQUIRED FOR EVERY 1000 SERVERS MANAGED BY SAS (GB)
Enterprise	2.0	3.1
Standard	1.5	3.1

See *Tablespace Sizes* in Chapter 2, on page 48.

For the disk space requirements of an upgrade, see the *Opsware® SAS Upgrade Guide*.

Hostname Setup

You need to be able to ping the database server hostname. To verify this, enter the following command:

```
ping <hostname>
```

or, on the database server, enter the following command:

```
hostname
```

If the hostname is not set up correctly, Oracle will not start up and you will encounter the following error:

```
ORA-00600: internal error code, arguments: [keltnfy-ldmInit],
[46], [1], [], [], [], [], []
```

Required Operating System Packages and Patches

The following sections list the packages and patches required by the Oracle 10g database. The Opsware Installer checks for these packages and patches before installing the Oracle database.

If you create the database using the Oracle Universal Installer rather than the Opsware Installer, you must check for these packages and patches manually.

Required Packages for RedHat Enterprise Linux AS3 32 bit x86

The following packages are required for Oracle 10g on Linux AS3 32 bit x86. These packages must be the versions listed or higher.

```
make-3.79.1
gcc-3.2.3-34
glibc-2.3.2-95.20
compat-db-4.0.14-5
compat-gcc-7.3-2.96.128
compat-gcc-c++-7.3-2.96.128
compat-libstdc++-7.3-2.96.128
compat-libstdc++-devel-7.3-2.96.128
openmotif21-2.1.30-8
setarch-1.3-1
libaio-0.3.96-5
```

Required Packages for RedHat Enterprise Linux AS4 64 bit x86

The following packages are required for Oracle 10g on Linux AS4 64 bit x86. These packages must be the versions listed or higher.

```
binutils-2.15.92.0.2-13.0.0.0.2.x86_64
compat-db-4.1.25-9.i386.rpm
compat-db-4.1.25-9.x86_64.rpm
compat-libstdc++-33-3.2.3-47.3.x86_64.rpm
compat-libstdc++-33-3.2.3-47.3.i386.rpm
control-center-2.8.0-12.x86_64.rpm
gcc-3.4.3-22.1.x86_64.rpm
gcc-c++-3.4.3-22.1.x86_64.rpm
glibc-2.3.4-2.9.i686.rpm
glibc-2.3.4-2.9.x86_64.rpm
glibc-common-2.3.4-2.9.x86_64.rpm
glibc-devel-2.3.4-2.9.x86_64.rpm
glibc-devel-2.3.4-2.9.i386.rpm
glibc-headers-2.3.4-2.9.x86_64.rpm
```



```
glibc-kernheaders-2.4-9.1.87.x86_64.rpm
gnome-libs-1.4.1.2.90-44.1.x86_64
libaio-0.3.103-3.i386.rpm
libaio-0.3.103-3.x86_64.rpm
libgcc-3.4.3-22.1.i386.rpm
libstdc++-3.4.3-22.1.x86_64
libstdc++-devel-3.4.3-22.1.x86_64
make-3.80-5.x86_64.rpm
pdksh-5.2.14-30.x86_64.rpm
sysstat-5.0.5-1.x86_64.rpm
xorg-x11-deprecated-libs-6.8.2-1.EL.13.6.i386.rpm
xscreensaver-4.18-5.rhel4.2.x86_64.rpm
```

To verify whether these rpms are installed on the OS, enter the following command:

```
rpm -q --qf '%{NAME}-%{VERSION}-%{RELEASE} (%{ARCH})\n'
<rpm_name>
```

Required Packages for Solaris 8, 9, and 10

Solaris 8, 9 and 10 must have the following packages:

```
SUNWarc
SUNWbash
SUNWbtool
SUNWhea
SUNWlibm
SUNWlibms
SUNWsprot
SUNWtoo
SUNWi1of
SUNWxwfnt
SUNWi1cs
SUNWsprox (only for Solaris 8 and Solaris 9)
SUNWi15cs
SUNWpool (only for Solaris 10)
SUNWpoolr (only for Solaris 10)
SUNWmfrun (only for Solaris 10)
```

Required Patches for Solaris 8

Solaris 8 must have the following patches (or later):

```
108528-23: SunOS 5.8: kernel update patch
108652-66: X11 6.4.1: Xsun patch
108773-18: SunOS 5.8: IIIM and X I/O Method patch
108921-16: CDE 1.4: dtwm patch
108940-53: Motif 1.2.7 and 2.1.1: Runtime lib. patch for
Solaris 8
108987-13: SunOS 5.8: Patch for patchadd and patchrm
```

```
108989-02: /usr/kernel/sys/acctctl & /.../exacctsys patch
108993-18: SunOS 5.8: LDAP2 client, libc, libthread ... lib.
patch
109147-24: SunOS 5.8: linker patch
110386-03: SunOS 5.8: RBAC Feature Patch
111023-02: SunOS 5.8: /kernel/fs/mntfs and ... sparcv9/mntfs
111111-03: SunOS 5.8: /usr/bin/nawk patch
111308-03: SunOS 5.8: /usr/lib/libmtmalloc.so.1 patch
111310-01: SunOS 5.8: /usr/lib/libdhcpagent.so.1 patch
112396-02: SunOS 5.8: /usr/bin/fgrep patch
111721-04: SunOS 5.8: Math Library (libm) patch
112003-03: SunOS 5.8: Unable to load fontset in 64-bit
Solaris 8 iso-1 or iso-15
```

Required Patches for Solaris 9

Solaris 9 must have the following patches (or later):

```
112233-11: SunOS 5.9: Kernel Patch
111722-04: SunOS 5.9: Math Library (libm) patch
```

Required Patches for Solaris 10

When Oracle 10.2 is installed on T2000 hardware with the Solaris 10 operating system, the Opsware Installer hangs during the installation of the Model Repository. The Oracle alert.log includes errors, such as the following:

```
MMNL absent for 28552 secs; Foregrounds taking over
Wed Aug  2 12:45:57 2006
MMNL absent for 28853 secs; Foregrounds taking over
Wed Aug  2 12:50:57 2006
MMNL absent for 29151 secs; Foregrounds taking over
```

Customers should look at Bug 6385446 from Sun Microsystems and apply Patches 118833-18, 119578-24 and 119254-24 as per:

`http://sunsolve.sun.com/search/document.do?assetkey=1-26-102289-1`

Opsware-Installed Oracle vs. a Standard Oracle RDBMS

An Oracle database created by the Opsware Installer differs in certain ways from a database installed using the Oracle Universal Installer, this section explains those differences.

Opsware Installer Changes to Database Configuration and Files

When the Opsware Installer installs the Oracle RDBMS software and creates the database, it makes the following changes:

- Creates the Unix user `oracle` locally in `/etc/passwd`.
- Creates the Unix groups `dba` and `oinstall` locally in `/etc/group`.
- Sets the `$ORACLE_HOME` environment variable to the following directory:

  ```
  /u01/app/oracle/product/10.2.0/db_1
  ```

- Sets the `$ORACLE_SID` environment variable to `truth`.
- Gets the service name (TNS name) from the Opsware Installer interview (truth.servicename prompt) and inserts it into the `tnsnames.ora` file in `$ORACLE_HOME/network/admin` and `/var/opt/oracle`. The Opsware Installer changes the value of the `host` parameter to the value returned by the Unix `hostname` command.
- Creates the data and index files under the following directories:

  ```
  /u01/oradata/truth
  /u02/oradata/truth
  /u03/oradata/truth
  ```

  The system administrator can configure the `/u01`, `/u02`, `/u03` directories before installing the Oracle RDBMS software.

- In the `/$ORACLE_HOME/network/admin/listener.ora` file, changes the value of the `host` parameter to the value returned by the Unix `hostname` command.

  The listener is password protected and OS authenticated. (The default password is `opsware`.) It listens on port 1521.

- Creates the `/etc/init.d/opsware-oracle` script, which you can use to start up and shut down the database and listener.

  This script is linked to corresponding scripts in the `/etc/rc*.d` directories.

- For Solaris 8 and 9, modifies `/etc/system` and asks the user to reboot the sever.
- For Solaris 10 and Linux, you are not required to reboot the server.

Database Parameter Value Differences

When it creates the Oracle database, the Opsware Installer sets the values for parameters in various files. This section lists the parameters set by the Opsware Installer that can be changed without adversely affecting Opsware SAS.

Kernel Parameter Differences in RedHat Enterprise Linux 3 AS and RedHat Enterprise Linux 4 AS

This section identifies the kernel parameters you can change for Linux 3 AS (32 bits) and Linux 4 AS (64 bits.

You can change values for the following parameters in `/etc/sysctl.conf`:

```
kernel.shmmax=2147483648
kernel.shmall=2097152
kernel.shmmni=4096
kernel.sem=256 32000 256 256 (for Linux 3 AS, 32 bits)
kernel.sem=250 32000 100 128 (for Linux 4 AS, 64 bits)
net.core.rmem_default=262144
net.core.wmem_default=262144
net.core.rmem_max=262144
net.core.wmem_max=262144
fs.file-max=65536f
net.ipv4.ip_local_port_range=1024 65000
```

You can change values for the following parameters in `/etc/security/limits.conf`:

```
oracle soft nofile 4096
oracle hard nofile 63536
oracle soft nproc 2047
oracle hard nproc 16384
```

You can change values for the following parameters in `/etc/pam.d/login`:

```
session required /lib/security/pam_limits.so (for Linux 3
AS, 32 bits)
session required pam_limits.so
```

Kernel Parameter differences in Solaris 8 and 9

The following parameters are set by the Opsware Installer in `/etc/system`:

```
forceload: sys/shmsys
forceload: sys/semsys
forceload: sys/msgsys
set shmsys:shminfo_shmmax=2147483648
set shmsys:shminfo_shmmin=1
set shmsys:shminfo_shmmni=100
```



```
set shmsys:shminfo_shmseg=10
set semsys:seminfo_semmns=2058
set semsys:seminfo_semmsl=256
set semsys:seminfo_semmni=100
set semsys:seminfo_semvmx=32767
set noexec_user_stack=1
```

You can change values for the following parameters in `/etc/system`:

```
set shmsys:shminfo_shmmin=1
set shmsys:shminfo_shmmni=100
set shmsys:shminfo_shmseg=10
set semsys:seminfo_semmns=2058
set semsys:seminfo_semmsl=256
set semsys:seminfo_semmni=100
set semsys:seminfo_semvmx=32767
set noexec_user_stack=1
```

You can increase the value for the following parameter in `/etc/system`:

```
set shmsys:shminfo_shmmax=2147483648
```

You can remove the following parameters in `/etc/system`:

```
forceload: sys/shmsys
forceload: sys/semsys
forceload: sys/msgsys
```

Kernel Parameter Differences in Solaris 10

To change a kernel parameter for Solaris 10, perform the following steps:

1. Enter set `noexec_user_stack=1` in `/etc/system`.
2. Run the following commands:

```
projadd -U oracle -K "project.max-shm-
memory=(priv,2048MB,deny)" user.oracle

projmod -s -K "project.max-sem-ids=(priv,100,deny)"
user.oracle
projmod -s -K "process.max-sem-nsems=(priv,256,deny)"
user.oracle

projmod -s -K "project.max-shm-ids=(priv,100,deny)"
user.oracle

echo "oracle::::project=user.oracle"  >> /etc/user_attr
```

3. Use the vi editor for `/etc/project` and `/etc/user_attr` to verify the changes made in step 2.

Differences in init.ora

You can increase values for the following parameters in `init.ora`:

```
db_cache_size=629145600
shared_pool_size=262144000
java_pool_size=52428800
large_pool_size=52428800
log_buffer=1048576
```

Location of Additional Oracle Data Files

If you want to add data files to a database created with the Opsware Installer, you can add them to the following directories:

```
/u01/oradata/truth
/u02/oradata/truth
/u03/oradata/truth
```

Pre-Oracle Universal Installer Tasks

If you create the database with the Opsware Installer, you do not need to perform the tasks in this section.

This section discusses the prerequisites for an installation of the Oracle RDBMS using the Oracle Universal Installer for use with Opsware SAS. For more detailed information about installing Oracle, see the *Oracle Installation Guide* for your operating system. Each operating system and Oracle version has a different guide. The Oracle documentation is available at the following URL:

`http://www.oracle.com/technology/documentation/index.html`

Before installing the Oracle RDBMS software, perform the following steps:

1. Verify that the server has the software listed in “Required Operating System Packages and Patches” on page 220.
2. Download and unzip the sample files.

   The sample files are available in the support area of the Opsware, Inc. web site at `www.opsware.com`. See “Sample Scripts and Configuration Files” on page 228.
3. Set the kernel parameters.

The easiest way to set these parameters is by copying and editing the following sample files:

```
kernel_params_redhat.txt
kernel_params_solaris.txt
```

These two files contain instructions, Unix commands, and lines of text for configuration files.

4. Create the required Unix users and groups by running the following commands. (If you use a directory different than `/u01/app/oracle`, modify the commands accordingly):

```
mkdir -p /u01/app/oracle
groupadd oinstall
groupadd dba
groupadd dboper
useradd -g oinstall -G dba \
  -d /u01/app/oracle -s /usr/bin/sh oracle
chown oracle:oinstall /u01/app/oracle
```

5. Set the environment variables for the `oracle` user.

   The easiest way to set these variables is by copying and editing the following sample files:

```
bash_profile
profile
```

Now you should be ready to install the Oracle RDBMS. For instructions, see the *Oracle Installation Guide* for your operating system.

Manually Creating the Oracle Database

If you create the database with the Opsware Installer, you do not need to perform the tasks in this section.

Sample Scripts and Configuration Files

Opsware, Inc. provides a bundle of sample files for you to copy and edit. Referenced throughout the instructions in this document, the sample files include SQL scripts, database configuration files, and kernel parameter settings.

The sample files are available in the support area of the Opsware, Inc. web site at `www.opsware.com`.

The following list summarizes the sample scripts and configuration files:

- **truth.sh**: A shell script that creates directories and then launches the `truth.sql` script.
- **truth.sql**: Prompts for passwords of the `SYS` and `SYSTEM` users and then launches the remainder of the SQL scripts in this list.
- **CreateDB.sql**: Creates a database with the UTF8 character set (as required by Opsware SAS), the data and index files, the default temporary tablespace, the undo tablespace, and the log files.
- **CreateDBFiles.sql**: Creates the following tablespaces that are required by Opsware SAS:

  ```
  LCREP_DATA
  LCREP_INDX
  TRUTH_DATA
  TRUTH_INDX
  AAA_DATA
  AAA_INDX
  AUDIT_DATA
  AUDIT_INDX
  ```

  See Table 2-6 on page 48 for additional tablespace sizing information.
- **CreateDBCatalog.sql :** Runs Oracle scripts to create data system catalog objects.
- **JServer.sql**: Sets up the Oracle Java environment.
- **CreateAdditionalDBFiles.sql**: Adds data and index files to certain tablespaces and allocates additional disk space. This script is optional, but recommended.

- **CreateUserOpsware_Admin.sql**: Creates the `opsware_admin` database user and grants permissions (privileges) to this user (required by Opsware SAS).
- **postDBCreation.sql:** Creates the `spfile` from the `pfile` (parameter file).
- **init.ora**: Contains initialization parameters for the database. See "Required and Suggested Parameters for init.ora" on page 230.
- **tnsnames.ora**: Enables resolution of database names used internally by Opsware SAS.
- **listener.ora**: Contains configuration parameters for the listener. Opsware SAS requires the listener to listen on port 1521.
- **bash_profile**: Sets environment variables and sets shell limits for the `oracle` Unix user.
- **profile**: Sets environment variables for the `oracle` Unix user.
- **kernel_params_redhat.txt**: Contains kernel parameters for RedHat Enterprise Linux 3 AS.
- **kernel_params_solaris.txt**: Contains kernel parameters for Solaris 8, 9, and 10.
- **opsware-oracle**: A script residing in `/etc/init.d` that starts up and shuts down the database and listener.

  Note that the `/etc/init.d/opsware-sas` script, which starts and stops the SAS components, does not start and stop the database and listener. For more information on the `opsware-sas` script, see the *Opsware® SAS Administration Guide*.
- **Export-Import**: A directory that contains parameter files and instructions for performing full database exports and imports.

Required and Suggested Parameters for init.ora

For Opsware SAS, the following `init.ora` entries are either suggested or required:

```
sga_max_size >=1GB
db_cache_size>=629145600
shared_pool_size>=262144000
java_pool_size>=52428800
large_pool_size>=52428800
log_buffer>=1048576
db_block_size>=8192
open_cursors >=300
session_cached_cursors=50
job_queue_processes >=10
nls_length_semantics=CHAR
nls_sort=GENERIC_M
processes >=1024
sessions >=1152
pga_aggregate_target >=104857600
workarea_size_policy=auto
change remote_login_passwordfile=SHARED
undo_management=AUTO (Suggested)
undo_tablespace=UNDO (Suggested)
query_rewrite_integrity=TRUSTED
query_rewrite_enabled=true
optimizer_mode=choose (for 9i) or all_rows (for 10g)
optimizer_index_cost_adj=20
optimizer_index_caching=80
cursor_sharing=SIMILAR, value can be set to
SIMILAR(preferred) or EXACT (recommended only if you
encounter Oracle Bug No. 3102053)
recyclebin=OFF (Suggested, for Oracle 10g only)
```

A bug in Oracle10g regarding DML containing inline views and certain types of subqueries causes Oracle to throw an ORA-00600 exception. Until the bug is fixed in Oracle 10g, the workaround is the following entry in `init.ora`:

```
_complex_view_merging = false
```

File Location Values in the Sample Scripts

In the sample scripts and configuration files, `ORACLE_HOME` environment variable is set to the following value:

```
/u01/app/oracle/product/10.2.0/db_1
```

The sample `init.ora` file has the following settings for files:

```
db_create_file_dest=/u01/oradata/truth
```



```
db_create_online_log_dest_1=/u02/oradata/truth
db_create_online_log_dest_2=/u03/oradata/truth

control_files=(/u02/oradata/truth/control01.ctl,/u03/
oradata/truth/control02.ctl)
```

If your organization has policies that do not match these settings, then you should modify the sample files accordingly.

Creating the Database with the Sample Scripts

To create the database with the sample scripts, perform the following steps:

1. Download and unzip the sample files.

   The sample files are available in the support area of the Opsware, Inc. web site at `www.opsware.com`. See "Sample Scripts and Configuration Files" on page 228.

2. Log in to the server as the Unix user `oracle`.

3. Copy the sample `init.ora` file to the following directory:
   `$ORACLE_BASE/admin/truth/create`

4. Examine the sample SQL scripts that you will run in step 6. If necessary, edit the scripts to conform to your organization's policies.

5. Log on to the server as the `oracle` user and change the mode of the sample `truth.sh` script:
   `chmod 755 truth.sh`

6. To launch the sample SQL scripts that create the database, run the `truth.sh` script:
   `./truth.sh`

7. After the scripts launched by `truth.sh` complete, check the log files in the following directory:
   `$ORACLE_HOME/assistants/dbca/logs`

Post-Create the Oracle RDBMS Tasks

If you create the database with the Opsware Installer, you do not need to perform the tasks in this section, except for step 1.

After creating the database, but before installing the Model Repository with the Opsware Installer, perform the following steps:

1. Create the `tnsnames.ora` file in the following directory:
   `$ORACLE_HOME/network/admin`
   Verify that the file conforms to the rules listed in "tnsnames.ora File Requirements" on page 233.
2. If it does not exist, create the following directory:
   ```
   mkdir -p /var/opt/oracle
   ```
3. Create the following symbolic link:
   ```
   ln -s $ORACLE_HOME/network/admin/tnsnames.ora \
   /var/opt/oracle/tnsnames.ora
   ```
4. Make sure that the oracle Unix user has read-write permission on the `tnsnames.ora` file.
5. For RedHat Enterprise Linux 3 AS, create another symbolic link:
   ```
   ln -s /etc/oratab /var/opt/oracle/oratab
   ```
6. Copy the sample `opsware-oracle` script to `/etc/init.d/`.
7. Link `/etc/init.d/opsware-oracle` to corresponding scripts in the `/etc/rc*` directories. For example:
   ```
   ln -s /etc/init.d/opsware-oracle \
         /etc/rc0.d/K02opsware-oracle
   ln -s /etc/init.d/opsware-oracle \
         /etc/rc1.d/K02opsware-oracle
   ln -s /etc/init.d/opsware-oracle \
         /etc/rc2.d/S60opsware-oracle
   ln -s /etc/init.d/opsware-oracle \
         /etc/rcS.d/K02opsware-oracle
   ```
8. Copy the sample `listener.ora` file to `$ORACLE_HOME/network/admin`.
9. In `listener.ora`, change the value of the `host` parameter to the host name of server running the database.

tnsnames.ora File Requirements

The `tnsnames.ora` file enables resolution of database names used internally by the core components. Opsware SAS has the following requirements for the `tnsnames.ora` file:

- The file must reside in the following location:

  `/var/opt/oracle/tnsnames.ora`

- If the core is installed across multiple servers, a copy of the file must reside on the servers running the following components:
  - Model Repository
  - Data Access Engine
  - Web Services Data Access Engine
  - Opsware Command Center
  - Global File System
  - Model Repository Multimaster Component
- For a core installed on multiple servers, the directory path of the `tnsnames.ora` file must be the same on each server.
- In a standalone core, the `tnsnames.ora` file must contain an entry for the Model Repository, as in the following example:

```
truth =
(DESCRIPTION=
(ADDRESS=(HOST=magenta.opsware.com)(PORT=1521)
(PROTOCOL=tcp))
(CONNECT_DATA=(SERVICE_NAME=truth)))
```

tnsnames.ora: Multimaster Mesh Requirements

In a multimaster mesh, the `tnsnames.ora` file must be set up for a central and a non-central core using the following guidelines.

Central (source, master) Core

The `tnsnames.ora` file must contain an entry for its own Model Repository. The port number must be set to the port that is used by the Oracle listener process, such as `1521`, `1526`, and so on.

The `tnsnames.ora` file must also contain an entry that specifies the central core Gateway. This port is used by the Data Access Engine for multimaster traffic. The port number is derived from the following formula: (20000) + (facility ID of the non-central core).

Example: In the following example, the TNS service name of the central core is `orange_truth`, which runs on the host `orange.opsware.com`. The TNS name of the non-central core is `cyan_truth`, which has a facility ID of `556`. Note that the entry for `cyan_truth` specifies `orange.opsware.com`, which is the host running the central core's Gateway.

```
orange_
truth=(DESCRIPTION=(ADDRESS=(HOST=orange.opsware.com)(PORT=1
521)(PROTOCOL=tcp))(CONNECT_DATA=(SERVICE_NAME=truth)))
cyan_
truth=(DESCRIPTION=(ADDRESS=(HOST=orange.opsware.com)(PORT=2
0556)(PROTOCOL=tcp))(CONNECT_DATA=(SERVICE_NAME=truth)))
```

Non-central (non-master) Core

The `tnsnames.ora` file must contain an entry for its own Model Repository. The port number must be set to that used by the Oracle listener process, such as `1521`, `1526`, and so on. The `tnsnames.ora` file does not require any entries for other cores in the mesh.

Example: In the following example, the TNS service name of the non-central core is `cyan_truth`, and the core runs on the host, `cyan.opsware.com`.

```
cyan_truth
=(DESCRIPTION=(ADDRESS=(HOST=cyan.opsware.com)(PORT=1521)(PR
OTOCOL=tcp))(CONNECT_DATA=(SERVICE_NAME=truth)))
```

Requirements for Enabling Oracle Daylight Saving Time (DST)

To enable Daylight Saving Time for the Oracle database, you must apply database tier patches. To apply these patches, perform the following steps:

1. Verify that your database is running on Oracle 9i or higher. If you are on an earlier database release, use one of the following MetaLink Notes to upgrade your database:

   10gR2 Database: MetaLink Note 362203.1

   9iR2 Database: MetaLink Note 216550.1

2. Use MetaLink Note 359145.1 to apply Oracle Database time zone fixes specific to your database version.

3 Use MetaLink Note 359145.1 to apply time zone fixes to the Oracle Java Virtual Machine (JVM) in the Oracle Database specific to your E-Business Suite database version.

Database Monitoring Strategy

Because the Model Repository is a critical component of Opsware SAS, the DBA should implement a monitoring strategy. The DBA can write custom monitoring scripts or use third-party products.

This section contains example commands for monitoring the Oracle database used by the Model Repository. When issuing the commands shown in this section, you must be logged on to the server as the user `oracle`:

```
$ su - oracle
```

The SQL commands shown in this section are entered in the `sqlplus` command-line utility. To run `sqlplus`, log on as `oracle` and enter the following command:

```
$ sqlplus "/ as sysdba"
```

Verify that the Database Instances are Up and Responding

To verify that the Database Instances are up and running, perform the following steps:

1 Check to see if the Oracle processes are running by entering the following command:

```
ps -ef | grep ora_
```

This `ps` command should generate output similar to the following lines:

```
oracle    1883     1  0 Jul24 ?        00:00:00 ora_pmon_truth
oracle    1885     1  0 Jul24 ?        00:00:00 ora_psp0_truth
oracle    1887     1  0 Jul24 ?        00:00:00 ora_mman_truth
oracle    1891     1  0 Jul24 ?        00:00:45 ora_dbw0_truth
oracle    1895     1  0 Jul24 ?        00:01:11 ora_lgwr_truth
oracle    1897     1  0 Jul24 ?        00:00:02 ora_ckpt_truth
oracle    1899     1  0 Jul24 ?        00:00:24 ora_smon_truth
oracle    1901     1  0 Jul24 ?        00:00:00 ora_reco_truth
oracle    1903     1  0 Jul24 ?        00:00:02 ora_cjq0_truth
oracle    2391     1  0 Jul24 ?        00:00:00 ora_qmnc_truth
oracle    2513     1  0 Jul24 ?        00:00:00 ora_q000_truth
oracle    2515     1  0 Jul24 ?        00:00:00 ora_q001_truth
oracle   18837     1  0 03:04 ?        00:00:00 ora_mmon_truth
oracle   18839     1  0 03:04 ?        00:00:00 ora_mmnl_truth
oracle   25184 24635  0 21:35 pts/1    00:00:00 grep ora_
```

2. Verify that the database status is `ACTIVE` by entering the following command in `sqlplus`:

```
select database_status from v$instance;
```

3. Verify that the open mode is `READ WRITE` by entering the following command in `sqlplus`:

```
select name, log_mode, open_mode from v$database;
```

Verify that the Datafiles are Online

To verify that the datafiles are online, in `sqlplus`, enter the following commands:

```
Col file_name format a50
Col status format a10
Set line 200
Select file_id, status, bytes, file_name from dba_data_files
order by tablespace_name;
```

The status should be AVAILABLE for all the data files.

Verify That the Listener is Running

To verify that the listener is running, perform the following steps:

1. Check to see if the Oracle listener processes are running by entering the following command:

```
ps -ef | grep tns

oracle    1762     1  0 Jul24 ?        00:00:01 /u01/app/
oracle/product/10.2.0/db_1/bin/tnslsnr LISTENER -inherit
oracle   25231 25189  0 21:39 pts/1    00:00:00 grep tns
```

2. Check the status of the listener with the `lsnrctl` command:

```
lsnrctl status
```

   The listener should be listening on port 1521 with the TCP protocol, and should be handling the instance named truth. The `lsnrctl` command should generate output similar to the following lines:

```
. . .
Connecting to (ADDRESS=(PROTOCOL=tcp)
(HOST=per1.performance.qa.opsware.com)(PORT=1521))
. . .
Instance "truth", status READY, has 1 handler(s) for this
service...
```

3. Test connectivity to the instance from the Data Access Engine (spin) and Web Services Data Access Engine (twist) hosts by running the `tnsping` utility:


```
tnsping truth
```

The OK statement displayed by the `tnsping` utility confirms that the listener is up and can connect to the instance. The `tnsping` utility should generate output similar to the following lines:

```
. . .
Used parameter files:

Used HOSTNAME adapter to resolve the alias
Attempting to contact (DESCRIPTION=(CONNECT_DATA=(SERVICE_
NAME=truth.performance.qa.opsware.com))(ADDRESS=(PROTOCOL=TC
P)(HOST=192.168.165.178)(PORT=1521)))
OK (0 msec)


Attempting to contact
(DESCRIPTION=(ADDRESS=(HOST=localhost)(PORT=1521)(PROTOCOL=t
cp))(CONNECT_DATA=(SERVICE_NAME=truth)))
OK (0 msec)
```

As an alternative to running the `tnsping` utility in this step, you can check the connectivity by running `sqlplus` and connecting to the database instance with the service name (TNS alias), for example:

```
sqlplus myuser/mypass@truth
```

Examine the Log Files

To examine the log files, perform the following steps:

1. Look for errors in the `alert.log` file.

   For each instance, locate the `alert.log` file in the background dump destination directory:

   ```
   $ORACLE_BASE/admin/<SID>/bdump
   ```

   Here is an example `bdump` directory for an instance with the `truth` SID:

   ```
   /u01/app/oracle/admin/truth/bdump
   ```

2. Look for errors in the other log and trace files, located in the following directories:

   ```
   $ORACLE_BASE/admin/<SID>/cdump
   $ORACLE_BASE/admin/<SID>/adump
   $ORACLE_BASE/admin/<SID>/udump
   ```

Check for Sufficient Free Disk Space in the Tablespaces

To check for sufficient disk space, perform the following steps:

1. Enter the following commands in `sqlplus`:

```
column dummy noprint
column  pct_used format 999.9       heading "Pct|Used"
column  name    format a16      heading "Tablespace Name"
column  Kbytes   format 999,999,999    heading "Current|File
Size|MB"
column  used    format 999,999,999   heading "Used MB "
column  free    format 999,999,999  heading "Free MB"
column  largest    format 999,999,999  heading
"Largest|Contigous|MB"
column  max_size format 999,999,999 heading "Max
Possible|MB"
column  pct_max_used format 999.999       heading
"Pct|Max|Used"
break   on report
compute sum of kbytes on report
compute sum of free on report
compute sum of used on report

select nvl(b.tablespace_name,
             nvl(a.tablespace_name,'UNKOWN')) name,
       kbytes_alloc Kbytes,
       kbytes_alloc-nvl(kbytes_free,0) used,
       nvl(kbytes_free,0) free,
       ((kbytes_alloc-nvl(kbytes_free,0))/
                         kbytes_alloc)*100 pct_used,
       nvl(largest,0) largest,
       nvl(kbytes_max,kbytes_alloc) Max_Size,
       ((kbytes_alloc-nvl(kbytes_free,0))/kbytes_max)*100
pct_max_used
from ( select sum(bytes)/1024/1024 Kbytes_free,
              max(bytes)/1024/1024 largest,
              tablespace_name
       from  sys.dba_free_space
       group by tablespace_name ) a,
     ( select sum(bytes)/1024/1024 Kbytes_alloc,
              sum(decode(maxbytes,0,bytes,maxbytes))/1024/
1024 Kbytes_max,
              tablespace_name
       from sys.dba_data_files
       group by tablespace_name
       union all
      select sum(bytes)/1024/1024 Kbytes_alloc,
              sum(decode(maxbytes,0,bytes,maxbytes))/1024/
1024 Kbytes_max,
              tablespace_name
       from sys.dba_temp_files
       group by tablespace_name) b
```



```
where a.tablespace_name (+) = b.tablespace_name
order by 1
/
```

In the output generated by the preceding commands, compare the numbers under the `Used` and `Free` headings.

2. To list the existing data, index, and temp files, enter the following commands in `sqlplus`:

```
Select file_id, bytes, file_name from dba_data_files;
Select file_id, bytes, file_name from dba_temp_files;
```

3. If a tablespace has auto-extended to its maximum size and is running out of disk space, then add new data files by entering the `ALTER TABLESPACE` command in `sqlplus`.

   The following example commands add data files to four of the tablespaces. For a full list of tablespaces and data files, see the output generated by the commands in the preceding two steps.

```
ALTER TABLESPACE "AAA_DATA"
ADD DATAFILE '/u01/oradata/truth/aaa_data10.dbf'
SIZE 32M AUTOEXTEND ON NEXT 128M MAXSIZE 4000M ;

ALTER TABLESPACE "AAA_INDX"
ADD DATAFILE '/u02/oradata/truth/aaa_indx11.dbf'
SIZE 32M AUTOEXTEND ON NEXT 128M MAXSIZE 4000M ;

ALTER TABLESPACE "UNDO"
ADD DATAFILE '/u03/oradata/truth/undo12.dbf' SIZE 32M
AUTOEXTEND ON NEXT 128M MAXSIZE 4000M ;

ALTER TABLESPACE "TEMP" ADD
TEMPFILE '/u04/oradata/truth/temp14.dbf' SIZE 32M AUTOEXTEND
ON NEXT 128M MAXSIZE 4000M ;
```

Verify That the Jobs in DBA_JOBS Ran Successfully

When the Model Repository is installed, the Opsware Installer sets up these jobs, which perform statistics and garbage collection. If these jobs do not run successfully, database performance will degrade.

To verify that the Jobs in DBA_JOBS ran successfully, perform the following steps:

1. To see if the jobs have run successfully, enter the following commands in `sqlplus`:

```
Col schema_user format a10
Col what format a50
Set line 200
Select job, schema_user, last_date, this_date, next_date,
broken, what from dba_jobs;
```

In the output generated from the preceding statement, the value of the "what" column indicates the type of job. If the value of "what" is `DBMS_STATS*` or `GATHER_*`, the job performs statistics collection. The jobs owned by 'GCADMIN' perform the garbage collection.

2. If you need to run the statistics and collection jobs manually, start by entering the following command in `sqlplus`:

```
grant create session to truth, aaa, lcrep;
```

To run the statistics collection jobs manually in `sqlplus`, enter `exec` commands similar to the example shown in this step.

If you copy and paste the following `exec` command examples, substitute the variables such as `schema_user_1` with the values of the `schema_user` column displayed by the preceding `select` statement. Substitute the variables such as `job_no_1` with the values of the `job` column displayed by the same `select` statement.

```
connect <schema_user_1>/<password>
exec dbms_job.run(<job_no_1>)

connect < schema_user_2>/<password>
exec dbms_job.run(<job_no_2>);

connect < schema_user_3>/<password>
exec dbms_job.run(<job_no_3>)

connect < schema_user_4>/<password>
exec dbms_job.run(<job_no_4>);
```

3. To run the garbage collection jobs manually, enter the following commands in `sqlplus`, substituting the job ID variables such as `job_no_1`:

```
grant create session to gcadmin;
connect gcadmin/<password_of_gcadmin>

exec dbms_job.run(<job_no_1>);
exec dbms_job.run(<job_no_2>);
exec dbms_job.run(<job_no_3>);
exec dbms_job.run(<job_no_4>);
```


4. If you entered the `grant` command in step 2, enter the following command in `sqlplus`:

```
revoke create session from truth, aaa, lcrep;
```

Monitor the ERROR_INTERNAL_MSG Table

The garbage collection jobs write exceptions to the `truth.ERROR_INTERNAL_MSG` table. Monitor this table daily for errors.

Monitor Database Users

To monitor database users, perform the following steps:

1. To check the database users, enter the following command in `sqlplus`:

```
Select username, account_status, default_tablespace,
temporary_tablespace from dba_users;
```

   The preceding `select` command should display the following users:

```
OPSWARE_PUBLIC_VIEWS
TRUTH
AAA_USER
LCREP
GCADMIN
TWIST
SPIN
AAA
OPSWARE_ADMIN
VAULT
```

   (The `VAULT` user is for multimaster databases only.)

   The `default_tablespace` of the Opsware SAS users should not be `SYSTEM` or `SYSAUX`. The `temporary_tablespace` of all users should be `TEMP`.

2. If a database user listed in the preceding step has the `account_status` of `LOCKED`, then unlock the user by entering the following command in `sqlplus`:

```
ALTER USER <username> ACCOUNT UNLOCK;
```

Troubleshooting System Diagnosis Errors

If an additional privilege (permission) has been made manually to the database, when Opsware SAS performs a system diagnosis on the Data Access Engine, an error message might be generated. For example, if an additional grant has been made to the `truth.facilities` table, the following error appears:

```
Test Information
Test Name: Model Repository Schema
Description: Verifies that the Data Access Engine's version
of the schema
matches the Model Repository's version.
Component device: Data Access Engine
(spin.blue.qa.opsware.com)
Test Results: The following tables differ between the Data
Access Engine and
the Model Repository:  facilities.
```

To fix this problem, revoke the grant. For example, if you need to revoke a grant on the `truth.facilities` table, log on to the server with the database and enter the following commands:

```
su - oracle
sqlplus "/ as sysdba"
grant create session to truth;
connect truth/<truth passwd>;
revoke select on truth.facilities from spin;
exit
sqlplus "/ as sysdba"
revoke create session from truth;
```

Garbage Collection

Opsware SAS creates four Oracle jobs for garbage collection or for deleting the old data. For details about how these jobs are set up, see the Oracle Jobs section of the Opsware SAS documentation.

By default, the garbage collection is run daily. The default values for retaining the data are as follows:

```
DAYS_WAY = 30 days
DAYS_TRAN = 7 days
DAYS_CHANGE_LOG = 180 days
DAYS_AUDIT_LOG = 180 days
```

These values can be read or updated in the `AUDIT_PARAMS` table. See Table A-6.

These values must be exactly the same for all the cores in a mesh.

To view the data, run the following sql command:


```
1* select name, value from audit_params
```

Table A-6: Garbage Collection Parameters

NAME	VALUE
DAYS_WAY	30
DAYS_TRAN	7
DAYS_CHANGE_LOG	180
LAST_DATE_WAY	07-OCT-06
LAST_DATE_TRAN	30-OCT-06
LAST_DATE_CHANGE_LOG	10-MAY-06
DAYS_AUDIT_LOG	180
LAST_DATE_AUDIT_LOG	10-MAY-06

To update the data, run a sql command similar to the following example as user lcrep:

```
update audit_params set value=x where name = 'DAYS_AUDIT_LOG';
```

These values must be exactly the same for all the cores.

Oracle Database Backup Methods

It is important that you back up the database on a regular basis. Be sure to use more than one backup method and to test your recovery process.

You can use the following methods to back up the Oracle database:

- **Export-Import**: An export extracts logical definitions and data from the database and writes the information to a file. Export-import does not support point-in-time recoveries. Do not use Export-Import as your only backup and recovery strategy.

  See the information on the `Export-Import` subdirectory in “Sample Scripts and Configuration Files” on page 228.

- **Cold or Off-Line Backups**: This procedure shuts the database down and backs up all data, index, log, and control files. Cold or off-line backups do not support point-in-time recoveries.

- **Hot or Online Backups**: During these backups, the database must be available and in ARCHIVELOG mode. The tablespaces are set to backup mode. This procedure backs up tablespace files, control files, and archived redo log files. Hot or online backups support point-in-time recoveries.
- **RMAN Backups**: While the database is either off-line or on-line, use the `rman` utility to back up the database.

Regardless of your backup strategy, remember to back up all required Oracle software libraries, parameter files, password files, and so forth. If your database is in ARCHIVELOG mode, you also need to back up the archived log files.

For more information on backing up Oracle databases, see the following documents:

- *Oracle Database 2 Day DBA*
- *Oracle Database Concepts*
- *Oracle Database Administrator's Guide*

These guides are on the Oracle web site at the following URL:

```
http://www.oracle.com/technology/documentation/index.html
```

Useful SQL

The following sql commands help you manage information in the Oracle database that the Model Repository uses.

Locked and Unlocked User

A user in Oracle 10.2.0.2 will be locked out after ten unsuccessful logons.

To verify whether the user has been locked or unlocked, enter the following sql command:

```
select username, account_status from dba_users;
```

To unlock the user, enter the following sql command:

```
>ALTER USER <username> ACCOUNT UNLOCK;
```

GATHER_SYSTEM_STATS

Sometimes the `GATHER_SYSTEM_STATS` job will be suspended. To remove this from 'AUTOGATHERING" mode, perform the following steps:

1. Select `PNAME, pval2` from `SYS.AUX_STATS$` where `pname ='STATUS';.`

2. If the PVAL2 status is "AUTOGATHERING", run GATHER_SYSTEM_STATS with gathering_mode=('STOP');.
3. Run your job 'exec dbms_job.run(xxx);.

BIN$ Objects

If the Opsware Installer discovers the existence of BIN$ objects in the database, enter the following sql commands:

```
show parameter recyclebin;
SELECT owner,original_name,operation,type FROM dba_
recyclebin;
connect <owner>/password
purge recyclebin; or purge table BIN$xxx;
```

By default, recyclebin is set to OFF.

Model Repository Installation on a Remote Database Server

To install or upgrade the Model Repository on a remote database server, perform the following steps:

1. Install the following on the server that will run the Opsware Installer:
   1. Full Oracle client or Oracle instant client, depending on the Opsware SAS version
   2. Set up the tnsnames.ora file to access the Truth/database
2. Set up the following on the Truth/database server:
   1. Log in as user oracle
   2. cd $ORACLE_HOME/network/admin
   3. Make sure that the listener.ora file has the following SID_LIST_* section:

```
SID_LIST_<your_listener_name> =
   (SID_LIST =
    (SID_DESC=
        (SID_NAME=truth)
        (ORACLE_HOME=<oracle_home>
```

   4. Make sure that the listener is started with the command:

```
lsnrctl start <your_listener_name>
```

Troubleshooting Model Repository Installation

When you install or upgrade the Model Repository on a remote database server, Oracle gives the following error and the Opsware Installer aborts:

```
Error: ORA-12526: TNS:listener: all appropriate instances are in
restricted mode
```

Problem

When Opsware SAS installs or upgrades the schema in the Oracle database, it puts the database in a "restricted mode". In Oracle 9i, users with "restricted session" privileges could connect to the remote database. In Oracle 10g, the standard listener will reject connections if the database is in a restricted mode. In Oracle 10g, a database administrator can only access the restricted instance locally from the machine that the instance is running on.

Solution

In Oracle10g, if the listener has the `SID_LIST_*` paragraph in the `listener.ora` file, then the users with "restricted session" privilege are able to connect to a remote database, even if the database is in restricted more. If the `listener.ora` file does not have the `SID_LIST_*` paragraph, then the listener rejects the client connections and gives an `ORA-12526: TNS: listener: all appropriate instances are in restricted mode` error.

Example: A listener.ora Entry

```
OPSCORE1 =
       (DESCRIPTION_LIST =
         (DESCRIPTION =
           (ADDRESS = (PROTOCOL = TCP)(HOST =
    opscore1.mycompany.com)(PORT = 1521))
           (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC0))
         )
       )

    SID_LIST_OPSCORE1 =
       (SID_LIST =
        (SID_DESC=
            (SID_NAME=truth)
            (ORACLE_HOME=/u01/app/oracle/product/10.2.0/db_1)
        )
         (SID_DESC =
           (SID_NAME = PLSExtProc)
           (ORACLE_HOME = /u01/app/oracle/product/10.2.0/db_1)
```



```
            (PROGRAM = extproc)
          )
)
```

In this example, the listener alias is `OPSCORE1`.

To start, stop, or check the status of the listener, enter the following commands:

`su-` Oracle to the truth box

To start the listener, enter `lsnrctl start opscore1`.

To stop the listener, `Lsnrctl stop opscore1`.

To check the status of the listener, enter `lsnrct status opscore1`.

Appendix B: TIBCO Rendezvous Configuration for Multimaster

IN THIS APPENDIX

This section discusses the following topics:

- TIBCO Rendezvous and Opsware SAS
- TIBCO Rendezvous Configuration

TIBCO Rendezvous and Opsware SAS

In a multimaster mesh, Opsware SAS uses the TIBCO Certified Messaging system to synchronize the Model Repositories in different facilities. This appendix provides reference information about the TIBCO configuration for multimaster.

The Opsware Installer automatically installs and configures TIBCO Rendezvous. By default, the installer configures the Rendezvous neighbors in a star topology, in which the source core is at the center. Unless you want another configuration, no further action is required by you.

TIBCO Rendezvous Configuration

This section explains how to add TIBCO routers and neighbors. For more information, see the following TIBCO Rendezvous documentation:

- *TIBCO Rendezvous Installation Guide*
- *TIBCO Rendezvous Concepts*

Running the TIBCO Rendezvous Web Client

To run the TIBCO Rendezvous web client, enter the following URL in a web browser:

```
http://<hostname>:7580
```

The <hostname> is the IP address or fully-qualified host name of the server running the Model Repository Multimaster Component (vault). The TIBCO Rendezvous General Information page appears.

Adding a TIBCO Router

To add a TIBCO router, perform the following steps:

1. Run the TIBCO Rendezvous web client.
2. From the Navigation pane, select **Configuration ➤ Routers**. The Routers Configuration page appears.
3. Make sure that your browser can resolve the host name so that the link in the Router Name field functions correctly.
4. In the Router Name field, enter a value. Usually, you enter the facility name for the router name.
5. Click **Add Router**. The new router appears in the table on the page.
6. In the Local Network column under Interfaces, click the number link for the router you just added. The Local Network Interfaces Configuration page appears.
7. Define a new network by entering the following data:
   1. In the Local Network Name field, enter the network name. In most cases, the network is given the same name as the facility name.
   2. In the Service field, set the service to 7500.
   3. Click **Add Local Network Interface**. The new local network appears in the table in the page.
8. Click the link for the new local network name. The Subject Configuration page appears.
9. In the Subject field, enter a greater-than symbol (>) and click **Import** and **Export**. (The greater-than symbol means “any.”) The greater-than symbol appears in the Import Subjects and Export Subjects tables in the page.
10. Repeat the previous steps for the other facilities in the multimaster mesh.

Adding a TIBCO Rendezvous Neighbor

To add a TIBCO Rendezvous neighbor, perform the following steps:

1. In the core Gateway properties file, add the following line:
   `opswgw.ForwardTCP=<port>:<remote_realm>:<remote_host>:7501`
   The <port> is derived from this formula: 10000 + remote_facility_ID. The <remote_realm> is the realm name of the core Gateway in the remote facility. The <remote_host> is the IP address of the server running the Model Repository Multimaster Component (vault) in the remote facility. In the following example, the remote facility ID, is 667, the realm name is LIME, and the IP address of the Model Repository Multimaster Component is 192.168.165.98:
   `opswgw.ForwardTCP=10667:LIME:192.168.165.98:7501`
2. Run the TIBCO Rendezvous web client. From the navigation pane, click Routers under Configuration. The Routers Configuration page appears.
3. In the Neighbor column of the table, click the number link for the router you added in the previous procedure. The Neighbor Interfaces Configuration page appears. You must define a neighbor for each facility in the multimaster mesh, except for the local facility.
4. In the Host field under the Remote Endpoint section, enter the host name of the server running the local core Gateway.
5. In the Port field under the Local Endpoint section, enter 7501.
6. In the Port field under the Remote Endpoint sections, set the port to the value derived from the following formula: 10000 + remote_facility_ID.
7. In the Router Name field under the Remote Endpoint section, enter the router name for the other facility.
8. For the Connection Type, select Normal Connection.
9. Click **Add Neighbor Interface**. The Local and Remote endpoints are added to the table in the page.

Verifying TIBCO Rendezvous Configuration

To see if the neighbor has connections to a facility, perform the following steps:

1. Run the TIBCO Rendezvous web client.
2. Click Connected Neighbors in the navigation pane. For each neighbor you defined for this facility, you should see links for the rvrd interface.

Appendix C: Opsware Gateway Properties File

IN THIS APPENDIX

This section discusses the following topics:

- Syntax of the Opsware Gateway Properties File
- Options for the opswgw Command

This appendix provides reference information about the settings in the properties file used by the Opsware Gateway.

Syntax of the Opsware Gateway Properties File

An Opsware Gateway properties file can have the following entries:

`opswgw.Gateway=name`

(Required) Set the name of the Opsware Gateway. This name must be unique in a Gateway network.

`opswgw.Realm=realm`

(Required) All Opsware Gateways operate in a named realm. A realm is an abstract name given to the collection of servers which are serviced by the Gateways in the realm. Realms can support an IP address space which may overlap with another realm. Realms are also used to define bandwidth utilization constraints on Opsware SAS functions in that realm.

`opswgw.Root=true | false`

Indicates that this Gateway should act as a root of the Gateway network. All Gateways in a root realm must be root Gateways. The default is false.

`opswgw.Daemon=true | false`

Daemonize the process. The default is false.

`opswgw.Watchdog=true | false`

Start an internal watchdog process to restart the Gateway in case a failure or a signal. A `SIGTERM` sent to the watchdog will stop the watchdog and Gateway processes. The default is false.

`opswgw.HardExitTimeout=seconds`

The number of seconds the main thread will wait (after a restart or exit request) for internal threads and queues to quiesce before a hard exit is performed.

`opswgw.LogLevel=INFO | DEBUG | TRACE`

Set the logging level. The `DEBUG` and `TRACE` produce a lot of output which will only be relevant to developers. The default is `INFO`.

`opswgw.LogFile=file`

The basename of the log file.

`opswgw.LogNum=num`

The number of rolling log files to keep.

`opswgw.LogSize=size`

The size in bytes of each log file.

`opswgw.TunnelDst=[lip1:]lport1[:crypto1],...`

Start up a tunnel destination listener. The tunnel listener can listen on a list of ports (a comma-separated list with no spaces.) If the port is prefixed with an IP, then the listener will only bind to that IP address. Examples: `2001, 10.0.0.2:2001, 2001:/var/foo.pem, 10.0.0.2:2001:/var/foo.pem`

`opswgw.TunnelSrc=rhost1:rport1:cost1:bw1[:crypto1],...`

Create a tunnel between this Gateway and the Gateway listening at `rhost1:rport1`. The link `cost1` and link bandwidth `bw1` must be set. The cost is a 32bit unsigned int, and bandwidth is in Kbits/sec (K=1024bits). (Additional tunnels are separated by commas.) Examples: `gw.foo.com:2001:1:0, gw.bar.com:2001:10:256:/var/foo.pem`

`opswgw.TunnelTCPBuffer=bytes`

Set the size TCP send and recv buffer to `bytes`. The system's OS must be configured to handle this value. View the Gateway's log file to see if the value given here will work on the current system.

`opswgw.ValidatePeerCN=true | false`

Indicates whether the peer CN is validated. The peer needs to be turned off during the installation of an untrusted Gateway. The default is true.

`opswgw.ProxyPort=[lip1:]lport1,[lip2:]lport2,...`

The SSL proxy listen port. If more than one proxy listen port is needed, add more using a comma separated list.

`opswgw.ForwardTCP=[lip1:]lport1:realm1:rhost1:rport1,...`

Create a static TCP port forward. Forward the local port `lport` to the remote service `rhost:rport`, which is in `realm`. A blank `realm` (e.g., `lport::rhost:rport`) means route to the root realm.

`opswgw.ForwardUDP=[lip1:]lport1:realm1:rhost1:rport1,...`

Create a static UDP port forward. Forward local port `lport` to remote service `rhost:rport`, which is in `realm`. If `realm` is blank (e.g., `lport::rhost:rport`) it means route to the root realm. (Warning: Some UDP services, such as DHCP, cannot be proxied in this manner.)

`opswgw.GWAddress=lhost`

Set the local host address (IP or name) that this Gateway uses to tell other components how to contact it. This value is used by the core to discover new core-side Gateways. It is also used to communicate the active list of Gateways that are servicing a realm to proxy clients (such as Agents) via the `X-OPSW-GWLIST` mime header.

`opswgw.IdentPort=[lip:]lport`

Start up an ident service listening on local port `lport`.

`opswgw.FinalizeTCPPortMap=true|false`

If true, remove the TCP source port from the ident port map immediately before the socket is closed. If false, the mapping persists until the port is reused. Warning: Only use false if you know what you are doing. The default is true.

`opswgw.FinalizeUDPPortMap=true|false`

If true, remove the UDP source port from the ident port map immediately before the socket is closed. If false, the mapping persists until the port is reused. Warning: Only use false if you know what you are doing. The default is true.

`opswgw.AdminPort=[lip:]lport[:crypto1]`

Start up an administration interface listening on local port `lport`, which is optionally bound to the local IP `lip`. If crypto is desired, then include a crypto specification file name.

`opswgw.ConnectionLimit=int`

The soft memory tuning limit of maximum number of connections.

`opswgw.OpenTimeout=seconds`

Only wait this many seconds for a remote `CONNECT` call to establish a remote connection.

`opswgw.ConnectTimeout=seconds`

Only wait this many seconds for the `connect()` to complete. If a timeout occurs, then an HTTP 503 message is returned to the client (via the ingress Gateway). The client will get this message if the `ConnectTimeout` plus the Gateway mesh transit delay is less than the `OpenTimeout`.

`opswgw.ReorderTimeout=seconds`

In the event of out-of-order messages (for a TCP flow), limit the amount of time to wait for messages (needed for reassembly) to arrive.

`opswgw.QueueWaitTimeout=seconds`

Maximum time that a tunnel message can wait at the head of an internal routing queue (while waiting for a tunnel to be restored).

`opswgw.LsaPublishRate=seconds`

Send the Link State Advertisements (LSAs) every X seconds.

`opswgw.LsaExtendRate=count`

Send an extended LSA for every count number of normal LSAs. Example: If `LsaPublishRate` is 10.0 seconds and `LsaExtendRate` is 30, then every 30 LSAs (about every 300 seconds) an extended LSA is published.

`opswgw.LsaTTLMultiple=float`

Set the TTL for LSAs to this number multiplied by the `LsaPublishRate`. Example: If `LsaPublishRate` is 10 seconds and `LsaTTLMultiple` is 3 then, the TTL for LSAs published by this Gateway is set to 30 seconds.

`opswgw.LsaExtendTTLMultiple=float`

Set the TTL for extended LSAs to this number multiplied by the `LsaPublishRate` and the `LsaExtendRate`. Example: If the `LsaPublishRate` is 15 seconds and the `LsaExtendRate` is 30 and the `LsaExtendTTLMultiple` is 8, then the TTL for extended LSA information is 3600 seconds (because 15 * 30 * 8 = 3600). One function of the in-memory database of the extended LSA information is to form the `X-OPSW-GWLIST` MIME header.

`opswgw.MaxRouteAge=seconds`

Discard the routes from the routing table that have not been refreshed within this number of seconds.

`opswgw.TunnelTimeoutMultiple=float`

This number, multiplied by the `LsaPublishRate`, gives the maximum time that a tunnel can be idle before it is garbage collected.

`opswgw.DoNotRouteService=host1:port1,host2:port2,...`

If a local client creates a proxy connection to `host:port`, then do not route the message; service it locally. This is used to handle certain services locally in the Gateway's current realm.

`opswgw.ForceRouteService=`
`host1:port1:realm1,host2:port2:realm2,...`

If local client creates a proxy connection to `host:port`, then force the message to route to realm.

`opswgw.HijackService=host1:port1,host2:port2,...`

If the local Gateway sees a connection to host:port via a tunnel, and the source realm is different than the local realm, then service the connection. Otherwise, let the message continue to its destination. This feature is useful for implementing transparent caches.

`opswgw.EgressFilter=tcp:dsthost1:dstport1:srchost1:srcrealm1,...`

If the local Gateway sees a `tcp` connection attempt to `dsthost:dstport` from `srchost1:srcrealm1`, then allow the connection. The implied default is to deny all connections. If you want to allow all traffic, then specify `*:*:*:*:*`. Watch out for shell quoting. It is common for an egress filter to only allow connections from the root realm. This can be expressed by leaving the `srcrealm` blank. Example: `tcp:10.0.0.5:22:172.16.0.5:` would allow tcp connections to 10.0.0.5, port 22, from 172.16.0.5 in a root realm.

`opswgw.IngressMap=ip1:name,ip2:name,...`

When sending an open message (and the `srcip` is in the ingress map), append (as metadata) the `ip:name` mapping to the open message. This allows a remote egress filter to use the `name` as the `srchost` instead of the `ip`. This feature supports the addition of a server to a farm without the need to add the server to many `EgressFilter` entries.

`opswgw.LoadBalanceRule=`
`tcp:thost:tport:mode:rhost1:rport1:rhost2:rport2, ...`

When receiving an open connection message for `thost:tport`, load balance the connection over real hosts `rhost1:rport1, rhost2:rport2` etc. The load balance strategy is defined by `mode`. There is currently only one mode: `STICKY`. This mode does sticky load balancing based on a hash of the source realm and `ip`. Remember to add an egress filter for `thost:tport`. You do not need to add egress filters for the targets. Load balancing is only for tcp connections.

`opswgw.LoadBalanceRetryWindow=seconds`

If an error occurs when using a load balanced target (e.g., `rhost1:rport1` above) then the target is marked `in-error`. This parameter controls how many seconds a Gateway will wait until it re-tries the target. If the target is missing (i.e., an `RST` is received upon the connection request) the load balancer will silently try to find a good target.

`opswgw.MinIdleTime=seconds`

The minimum number of seconds a connection can be idle, during an overload condition, before it will be considered for reaping.

`opswgw.GCOverloadTrigger=float`

The fraction of `SoftConnectionLimit` at which to start overload protection measures. When the number of open connections hits this overload trigger point, the overload protection kicks in, reaping the most idle connections over `MinIdleTime`. Overload protection quits when the connection count falls below the overload trigger point.

`opswgw.GCCloseOverload=true | false`

When a client tries to open a connection after the `ConnectionLimit` has been reached, this property tells the Gateway what to do with the new connection. A value of true causes the Gateway to close the new connection. A value of false causes the Gateway to park the new connection in the kernel's backlog and to service it after the overload condition subsides. The proper setting is application dependent. The default is false.

`opswgw.VerifyRate=seconds`

When a connection stops moving data for this number of seconds, a connection verify message is sent to the remote Gateway to check that the connection is still open on its end. This check is repeated periodically and indefinitely when the timeout has expired.

`opswgw.OutputQueueSize=slots`

The size of the tunnel output queues. These queues store messages destined for remote Gateways. Each remote Gateway has an output queue.

`opswgw.DefaultChunkSize=bytes`

The default (maximum) IO chunk size when encapsulating a TCP stream. This default is only used on links with no bandwidth constraint.

`opswgw.LinkSaturationTime=seconds`

On links with a bandwidth constraint, the chunk size (see `DefaultChunkSize`) is computed based on two parameters. The first is the link's bandwidth constraint. The second is the amount of time that the bandwidth shaper should utilize the full, real, bandwidth on the link. This parameter controls the duty cycle of the bandwidth shaper. Smaller values give a smoother bandwidth control at the cost of more overhead, because each smaller IO chunk has a header.

`opswgw.MaxQueueIdleTime=seconds`

The maximum time to keep an idle output queue before garbage collection removes it.

`opswgw.TunnelPreLoad=slots`

The maximum number of output queue slots to use before waiting for the first `Ack` message. This allows for pipelining in Long Fat Pipes. This value is reduced geometrically to one as the number of queue slots diminish.

`opswgw.BandwidthAveWindow=samples`

The maximum number of IO rate samples for the bandwidth estimation moving window. The samples in this window are averaged to provide a low pass estimate of the bandwidth in use by a tunnel. This estimate has high frequency components due to the sharp edge of the filter window.

`opswgw.BandwidthFilterPole=float`

The pole of a discrete-time first-order smoothing filter used to remove the high frequency components of the moving window estimator. Set the value to 0.0 to turn off this filter.

`opswgw.StyleSheet=URL`

Add a stylesheet link to URL when rendering the admin UI. This is useful for embedding the admin UI in another web-based UI. In addition to using this property to control the default stylesheet, a dynamic stylesheet override is supported by adding the variable `StyleSheet=¡url¿/style.css` to the admin UI URL.

`opswgw.PropertiesCache=file`

Link cost and bandwidth can be controlled via parameter-modify messages over the tunnel connections. These real-time adjustments are made to the running process and written to a parameter cache which will override the properties file or command line arguments.

Options for the opswgw Command

All of the properties in the preceding section can be specified as options for the `opswgw` command. For example, the `opswgw.Gateway=foo` entry in the properties file is equivalent to the following command-line option:

```
/opt/opsware/opswgw/bin/opswgw --Gateway foo
```

Command-line arguments override corresponding entries in the properties file. In addition to the entries listed in the preceding section, the `opswgw` command can specify a properties file as follows:

```
/opt/opsware/opswgw/bin/opswgw --PropertiesFile file
```


Index

G

H

I

L

M

N

O

P

R

S

T

U

V

W

