Oracle® Communications Services Gatekeeper

Multi-tier Installation Guide

Release 7.0

E95426-01

July 2018

ORACLE®

Oracle Communications Services Gatekeeper Multi-tier Installation Guide, Release 7.0

E95426-01

Contents

6 Services Gatekeeper Post-Installation Tasks

7 Configuring the Services Gatekeeper Domain

Preface

This book explains how to install Oracle Communications Services Gatekeeper. It includes instructions for WebLogic Server domain configuration and post-installation tasks.

Audience

The person installing the software should be familiar with the following topics:

- Operating system commands
- Database configuration
- WebLogic Server

Before reading this guide, you should have a familiarity with Services Gatekeeper. See *Services Gatekeeper Concepts*.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

For related information, see the following Services Gatekeeper documents:

- *Oracle Communications Services Gatekeeper Concepts*
- *Oracle Communications Services Gatekeeper Licensing Guide*
- *Oracle Communications Services Gatekeeper Security Guide*
- *Oracle Communications Services Gatekeeper System Administrator's Guide*

For related information about Oracle WebLogic Server 11*g* or 12*c*, see the following documents:

- *Oracle Fusion Middleware Introduction to Oracle WebLogic Server*

- *Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server*
- *Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server*
- *Oracle Fusion Middleware Using Clusters for Oracle WebLogic Server*
- *Oracle Fusion Middleware Securing Oracle WebLogic Server*

The Oracle WebLogic Server 11*g* and 12*c* documentation are available on the Oracle WebLogic Server Product Documentation page of the Oracle Technology Network website:

http://www.oracle.com/technetwork/middleware/weblogic/documentation/index.html

1

Services Gatekeeper Installation Overview

This chapter describes the general process of installing and configuring Oracle Communications Services Gatekeeper.

To learn more about installing WebLogic Server products in general, and about the installer program in particular in regard to WebLogic Server, see *Installing and Configuring Oracle WebLogic Server and Coherence Software* at the Oracle Fusion Middleware Documentation website.

Overview of Installed Components

During the installation process, you install and configure the following components:

- Your database
- The Services Gatekeeper software, which includes:
  - Container services and communication services applications
  - Portal Server
  - Platform Development Studio (PDS)
  - WebLogic Server, which is the platform container
- (Optional) Services Gatekeeper Application Test Environment (ATE)
- (Optional) Services Gatekeeper Platform Test Environment (PTE)
- (Optional) Services Gatekeeper Reports

Overview of the Services Gatekeeper Installation Procedure

The installation procedure follows these steps:

1. Plan your installation. When planning your installation, do the following:
   - Determine the scale of your implementation, for example, a small development system, or a large production system.
   - Determine how many physical machines you need, and which software components to install on each machine.
   - Plan the system topology, for example, how the system components connect to each other over the network.
2. Review system requirements. System requirements include:
   - Hardware requirements, such as processor and disk space.

- System software requirements, such as operating system (OS) versions and OS patch requirements.
- Information requirements, such as IP addresses and host names.

3. Perform pre-installation tasks such as installing the database and the JDK.
4. Install Services Gatekeeper.

   When you install Services Gatekeeper, WebLogic Server is also installed.
5. Perform post-installation tasks:
   - Configure your domains.
   - Install optional Services Gatekeeper components, such as Reports, Application Test Environment, and Platform Test Environment.
   - Perform additional configuration tasks.
6. Verify the installation.
7. If you are upgrading from a previous version of Services Gatekeeper, perform the upgrade tasks. See "Upgrading Services Gatekeeper" for more information.

After Services Gatekeeper is installed, perform some system administration tasks such as the following tasks, among others:

- Configure system security, including user names and passwords.
- Configure container services and communication services.
- Set up service provider and application accounts and service level agreements (SLAs).

Services Gatekeeper Installation Options

When installing Services Gatekeeper, you can use the following types of installation:

- Single-tier installation

  The single-tier (default) installation installs a preconfigured Services Gatekeeper implementation that includes everything you must run a test system, or a small to medium size production environment on a single system. This is the fastest installation option and also appropriate for test and evaluation systems. It includes a Services Gatekeeper Administration Server, managed server (with integrated network and access tiers), and a Java DB database.

  The single-tier installation supports GUI mode only.
- Multi-tier installation

  A multi-tier installation is appropriate for deployments where the Access Tier and Network Tier servers are in separate clusters. There are two installer modes for multi-tier installations:
  - GUI mode: An interactive mode that uses a graphical user interface
  - Silent mode: A non-interactive mode that uses a script and an XML input file

    Silent mode is a way of setting installation options once and then using those settings to duplicate the installation on many systems. The installation program reads your settings from a file that you create prior to beginning the installation. The installation program does not display any options during the installation process. Silent-mode installation works on Windows, Linux, and Solaris systems.

Ensuring a Successful Services Gatekeeper Installation

The Services Gatekeeper installation should be performed only by qualified personnel. You must be familiar with Oracle WebLogic Server and the supported operating systems. You should be experienced with installing Java-related packages. Oracle recommends that the Oracle database installation and configuration be performed by an experienced database administrator.

Follow these guidelines:

- As you install each component (for example, the Oracle database), verify that the component installed successfully before continuing the installation process.
- Pay close attention to the system requirements. Before you begin installing the software, make sure your system has the required base software. In addition, make sure that you know all of the required configuration values, such as host names and port numbers.
- As you create new configuration values, write them down. In some cases, you might need to re-enter configuration values later in the procedure.

Installer Version Information

Two JAR files, **ocsg_version.jar** and **ocsg_patch_version.jar** are included in the directory **`${OCSG_INSTALL_DIR}/ocsg/server/lib/wlng`** to record the current Services Gatekeeper installer and patch version information. In addition, in the **`MANIFEST.MF`** files for these two jar files, the attribute `Implementation-Version` contains the following version information in the following format:

```
7.0.0.0.0 425 Wed Sep 27 23:49:22 PDT 2017 247576
```

Table 1–1 describes the version information:

Table 1–1 Version Information

Element (example only)	Description
7.0.0.0.0	Official release number
425	The load number on the build server
Wed Sep 27 23:49:22 PDT 2017	The timestamp consisting of the day of the week, the month, the date, the hour, minute, and second, and timezone.
247576	The revision number in the source control system

The two JAR files are packaged together in the installer. Within the Services Gatekeeper patch file, only **ocsg_patch_version.jar** is included in the patchset zip file. The **ocsg_version.jar** file is always excluded.

In a Services Gatekeeper installation with no patches, the **ocsg_patch_version.jar** and **ocsg_version.jar** files are the same; the version information shows only the installer version. If the installer has been patched, the **ocsg_patch_version.jar** file is updated and the version information shows both the base installer version and latest patchset version.

For an upgrade from one release version to another, the version information is that of the new Services Gatekeeper installer after doing the upgrade.

Obtaining the Version Information

You can obtain the Services Gatekeeper version information in the following ways:

- From the **default.log** file
- From the API management REST interface
- From the DAF MBean
- From the managed server's console

Using the REST Interface

The following example illustrates a REST Get request:

```
GET /prm_pm_rest/services/prm_pm/services/partner_manager/
sysconfig/installerVersion
```

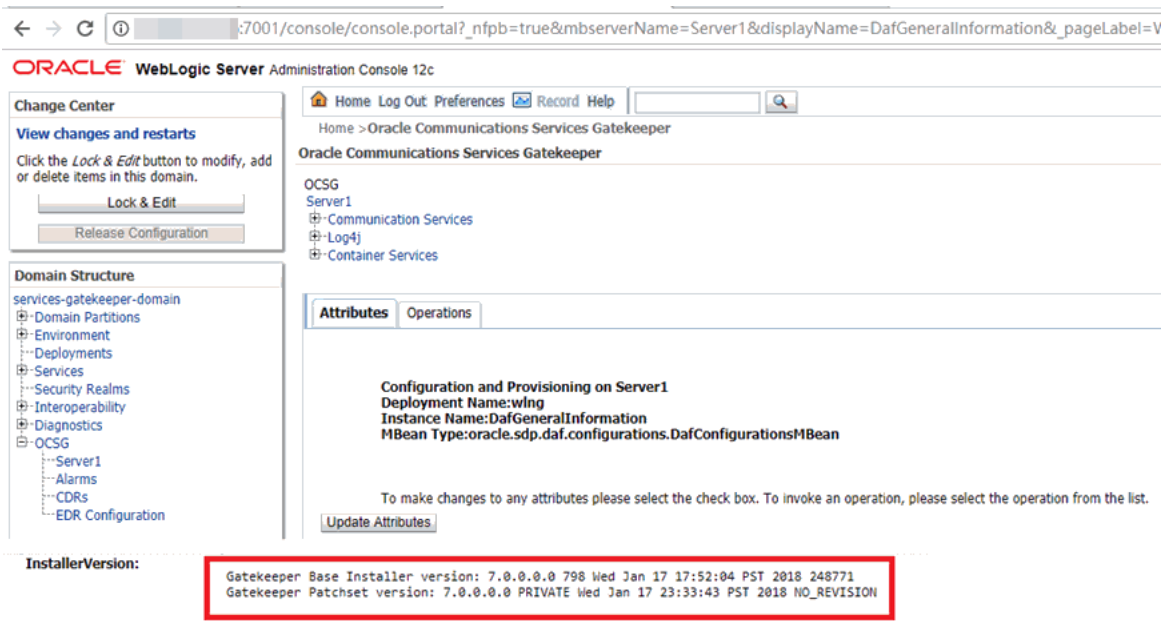
Authentication uses basic authentication with the partner manager's account and password. The response content is of type JSON as shown in the following example, which shows that a patch has been applied:

```
{
"version":"7.0.0.0.0 PRIVATE Mon Dec 25 07:37:38 PST 2017 NO_REVISION",
"patchsetVersion":"7.0.0.0.0 PRIVATE Mon Dec 25 12:12:00 PST 2017 NO_REVISION"
}
```

Using the DAF MBean

Figure 1–1 illustrates the Administration console display produced by an MBean request:

Figure 1–1 DAF MBean Result



On the managed server console, the output looks like this when no patches have been applied:

```
Gatekeeper base installer version is 7.0.0.1.0 861 Mon Mar 5 04:51:58 PST 2018
249229
```

If the installer has been patched, it looks like this:

```
Gatekeeper base installer version is 7.0.0.1.0 861 Mon Mar 5 04:51:58 PST 2018
249229

Gatekeeper patchset version is 7.0.0.2.0 891 Mon Mar 25 09:53:08 PST 2018 249300
```

Placeholders Used in this Guide

Table 1–2 shows the placeholders used in this guide:

Table 1–2 Placeholders Used Throughout Documentation

Placeholder	Description
Middleware_home	The directory that serves as the repository for common files that are used by Oracle Communications products installed on the same system, such as Services Gatekeeper and WebLogic Server. The files in the *Middleware_home* directory are essential to ensuring that software operates correctly on your system. They: ■ Facilitate checking of cross-product dependencies during installation ■ Facilitate Service Pack installation
Services_Gatekeeper_home	The directory in which the Services Gatekeeper software is installed. By default, this is a subdirectory of *Middleware_home*; for example, *Middleware_home***/ocsg**.
domain_home	The directory in which the Services Gatekeeper domain resides, located in *Middleware_home***/user_projects/domains**.
installer_file	The product installation file that you download and run to install the software.

2

Planning Your Services Gatekeeper Installation

This chapter provides information about planning your Oracle Communications Services Gatekeeper installation.

About Services Gatekeeper Software Components

Services Gatekeeper is built on top of Oracle WebLogic Server and can use all WebLogic Server components. It also embeds Oracle Communications Converged Application Server for connectivity to SIP networks and access to network nodes using the Diameter protocol.

About Services Gatekeeper Components for a Telecom Implementation

Services Gatekeeper provides communication services that telecom operator in-house applications and third-party applications use to access assets in the telecom network. For a list of the supported communication services, see *Services Gatekeeper Communication Service Reference Guide*. In addition, Services Gatekeeper provides extension points and tooling that you can use to create new communication services.

Each communication service has two components:

- A service facade that exposes interfaces to be used by applications.
- A service enabler that consists of a network protocol plug-ins. The plug-ins can be instantiated. Each instance connects to a node in the telecom network using a specific protocol. These instances interact with container services as necessary.

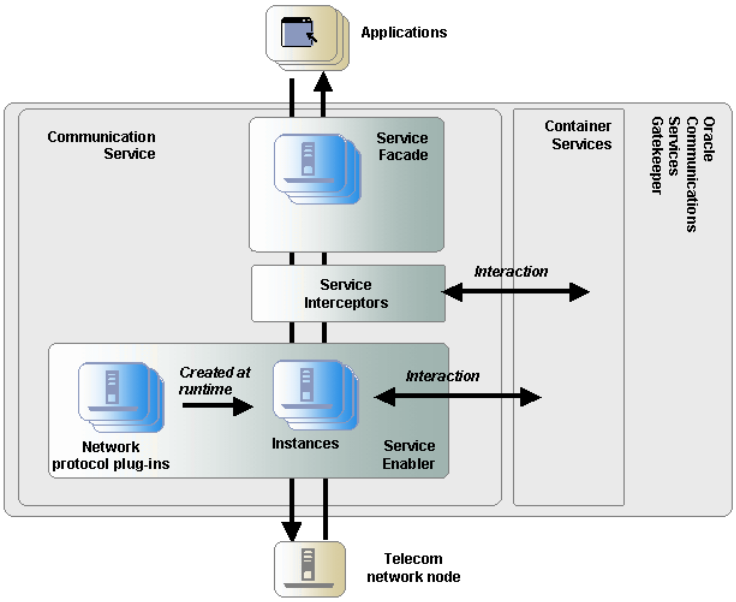
The communication services use container services, which are provided with the installation. Container services include Alarm, Event Data Record (EDR), Call Details Record (CDR), Policy Enforcement, Service Level Agreement (SLA) enforcement, Account, Event channel, and Trace services.

Requests between the network and applications can be intercepted by using service interceptors, which may allow, deny, or manipulate the request as necessary. When called upon to act on a request, service interceptors interact with container services as necessary to determine how to handle the request.

Service facades, service enablers, and service interceptors are deployable units in Services Gatekeeper.

Figure 2–1 illustrates how the Services Gatekeeper service components mediate the flow of requests between applications and the telecom network node.

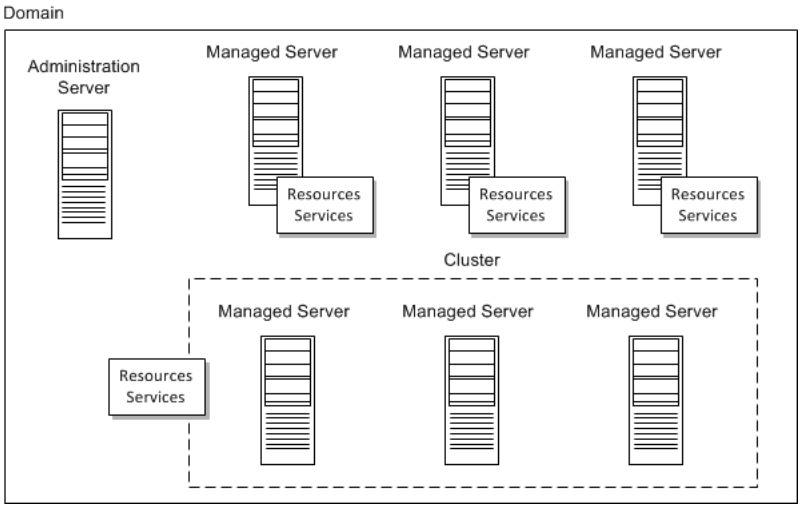
Figure 2–1 Services Gatekeeper Components



Understanding Services Gatekeeper Domains

A domain is the basic administrative unit in Oracle WebLogic Server. It consists of an Administration Server and, usually, one or more managed servers, which may be grouped into clusters, as illustrated in Figure 2–2.

Figure 2–2 WebLogic Domain



The Administration Server is used to manage the domain and provides access to the WebLogic Server administration tools.

A single WebLogic Server instance can function as both the Administration Server and a managed server, depending on the purpose of the installation. For example, developers creating communication service extensions using the Platform Development Studio might run both the Administration Server and managed servers on a single machine.

Managed servers are often grouped together into clusters that work together to provide scalability and high availability. Clusters improve performance and provide failover should a server instance become unavailable. The servers within a cluster can run on the same machine, or they can reside on different machines. To the client, a cluster appears as a single WebLogic Server instance.

Managed servers, or the clusters to which they are linked, host application components—in this case, the communication services—and resources, which are also deployed and managed as part of the domain.

Each server instance is also assigned to a machine that is a logical representation of actual hardware. The machine representation is used by the Administration Server to start and stop remote servers using the node manager. Multiple server instances can run in a single machine.

For more information about WebLogic Server domains, see "WebLogic Server Domains" in the WebLogic Server documentation.

See "About Configuring Domains" for information on configuring domains.

Understanding Services Gatekeeper Deployment Types

Services Gatekeeper supports the following types of deployments:

- Single-tier deployments, which are self-contained Services Gatekeeper implementation that runs on a single system. This is what most new customers use to become familiar with Services Gatekeeper. This implementation is suitable for test and development and small-scale production environments. See “Getting Started with Services Gatekeeper” in *Services Gatekeeper Getting Started Guide* for details.
- Tiered deployments, which are suitable for large production environments
- Non-tiered deployments, which are suitable for test and development and small-scale production environments. There are two types of non-tiered deployments:
  - basic developer
  - basic high availability
- Geographically redundant deployments, which are suitable for large production environments in which provisioning and run-time processing data are replicated between sites

Table 2–1 provides a summary of the different deployment types.

Table 2–1 Summary of Deployments

Deployment Type	Provides	Characteristics
Tiered	Access and Network clusters	Targeted for medium and large deployments. Some high-availability aspects. High level of security. High level of scalability.
Non-tiered	Basic developer configuration	Targeted to extension and integration developers. No high-availability or redundancy aspects.
Non-tiered	Basic high-availability	Targeted for smaller deployments, testing, and development. Introduces support for high availability or redundancy.
Geographic redundancy	Geographically separated sites with data synchronization	Each site has the characteristics of a tiered deployment. Adds geographic redundancy aspects that allow for disaster failover in the event of a site failure. Both sites are active; assumes site affinity from an application's point of view.

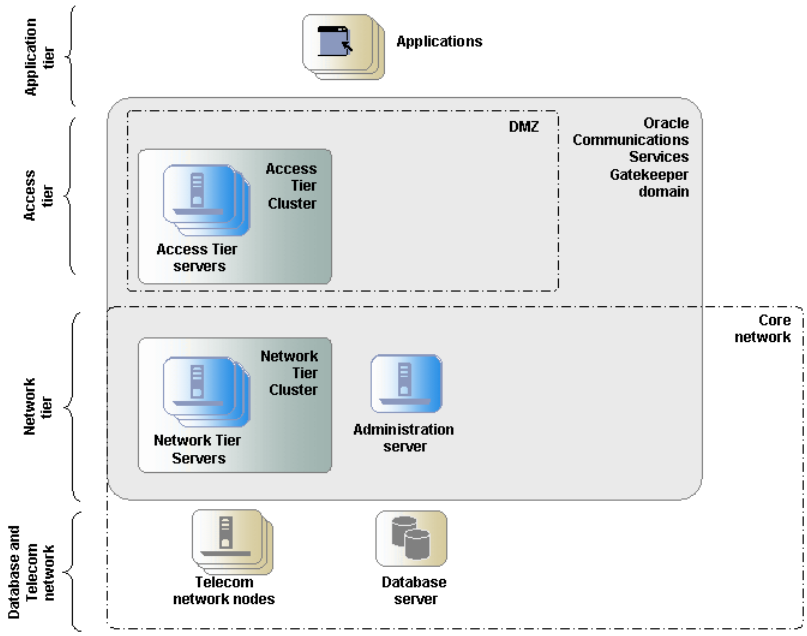
About Tiered Deployments

A Services Gatekeeper deployment is normally divided into three tiers: the Access Tier, the Network Tier, and the database tier.

As Figure 2–3 shows, in-house and third-party applications that use Services Gatekeeper interact only with the Access Tier. The Network Tier interacts with the telecommunications network, the Access Tier, and other nodes such as Operation Support Systems (OSS) or Business Support Systems (BSS).

Service facades are deployed in the Access Tier nodes. Service enablers and container services are deployed in the Network Tier nodes.

Figure 2–3 Example of a Tiered Deployment



The tiering physically separates the servers, which enables carrier-grade scaling, security, and high availability.

Services Gatekeeper uses storage services that rely on an underlying database. For security reasons and in order to scale the database tier independently, the database is normally running on separate, dedicated nodes.

Physical Architecture

Each deployment consists of a number of nodes to ensure high availability and to provide redundancy and load balancing. The nodes are separated into a tiered architecture.

Production deployments of Services Gatekeeper are normally tiered into an Access Tier, a Network Tier, and a database tier.

The Access Tier is responsible for:

- Security
- SSL (secure sockets layer) termination
- XML serialization
- Termination of more latent WAN connections with applications

The Network Tier is responsible for:

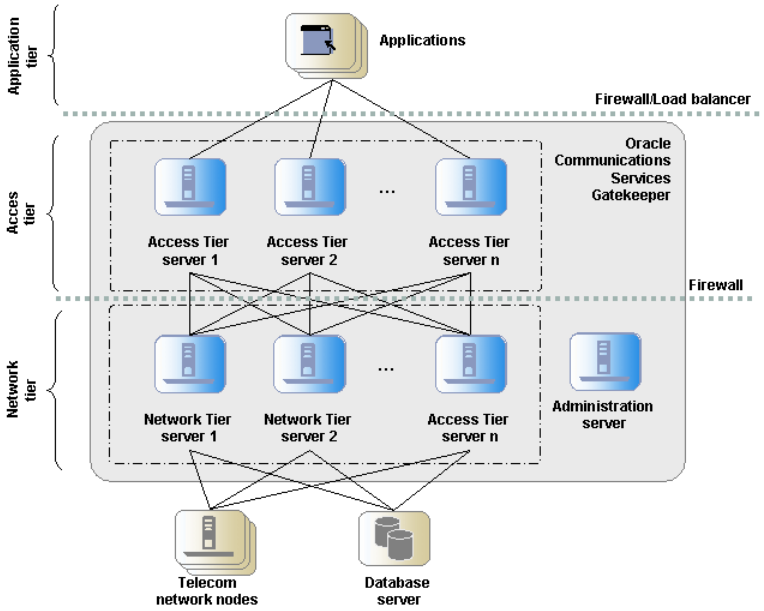
- Network protocol translation
- Generation of EDRs (event data records), CDRs (charge data records), and alarms
- Inter-node communications and state management

The database tier is a relational database management system (RDBMS) used to store configuration and provisioning data, as well as data generated as a result of interaction with applications and network nodes. The RDBMS is abstracted by the Storage Service, a container service that provides the nodes with access to a shared cache. The Storage Service is deployed on nodes in the Network Tier.

Figure 2–4 shows an example of the tiered deployment of servers in a Services Gatekeeper installation. A firewall and a load balancer separate the application tier from the servers in the Access Tier. A second firewall regulates the traffic between the servers in the Access Tier and the servers at the Network Tier. All servers in the Network Tier are managed by the Services Gatekeeper Administration Server. The managed servers in the Network Tier can access the telecom network nodes and the database servers.

See "Deploying Services Gatekeeper in a Demilitarized Zone" in *Services Gatekeeper Security Guide* for more information on setting up and protecting the Services Gatekeeper physical architecture.

Figure 2–4 Servers in a Tiered Deployment



As seen in Figure 2–4, the tiers provide a separation at the network level, which allows for:

- Firewalls to be introduced between the tiers
- Different networks to be used for the different tiers

The servers are also physically separated within a tier.

Each tier consists of at least one cluster, with at least two server instances (nodes) per cluster, and all server instances run in active mode, independently of each other. The servers in all clusters are, in the context of Oracle WebLogic Server, managed servers. Together the clusters make up a single WebLogic Server administrative domain, controlled through an Administration Server.

Runtime Aspects

Nodes are grouped into one or more clusters in a deployment. With a few exceptions, all the same components are deployed on nodes within a cluster and these components are managed as one unit. There are a set of services where a component, a cluster singleton, is active only on one node within the cluster at any given point. In case of node failure, the singleton service is automatically migrated to another node in the cluster.

Configuration settings for a deployed module can be per node or shared among the components deployed in the a cluster.

The clusters are grouped into a domain, with an Administration Server that normally does not process any traffic. The traffic-processing nodes are called managed servers.

There is normally a one-to-one relationship between managed servers and a physical servers. In some cases, several managed servers can run on the same physical server. If the CPU is powerful and has a lot of memory, performance can benefit from using a smaller heap size for a set of managed servers on a single physical server, rather than using one managed server per physical server. The disadvantage of this setup is mainly loss of redundancy and lower availability if a physical server fails.

Scalability

The Access Tier and the Network Tier scale independently of each other. New servers can be added at runtime, allowing you to scale the deployment horizontally.

The main responsibilities of the Access Tier are security, SSL termination, XML serialization and termination of WAN connections from applications. The main responsibilities of the Network Tier are to perform network protocol translation and protocol abstraction. This separation of responsibilities translates into two very different processing models.

Processing in the Access Tier is CPU-intensive, mainly concerned with XML to Java translations that have a short life span and generate numerous short-lived objects that trigger frequent garbage collection. The Access Tier does not maintain any state information. This behavior is consistent across communication services.

Processing in the Network Tier maintains state information and puts demands on data caching, inter-node communication, and processing logic.

The behavior of communication services varies with some of the services supporting relatively long-lived sessions. Call control sessions tend to have a significantly longer session lifetime than more data-centric sessions, such as messaging.

Some protocols, for example short message peer-to-peer protocol (SMPP), have more efficient data transfer sizes compared to XML-based protocols, for example multimedia messaging service interface (MM7). This translates into different processing needs.

Sizing and configuring individual servers in the Network Tier depend on which communication services are used in the deployment and the estimated utilization ratio between them. Both the physical characteristics of the servers (such as internal memory, network cards and CPU speeds) and settings for the Java Virtual Machine, (such as heap size and other parameters that affect garbage collection) can be optimized for the different use cases.

In summary, the processing models determine how you optimize the physical hardware, the Java Virtual Machine, and the operating system. Tiering allows you to tune individual nodes in each tier for the different processing requirements.

Securing the Physical Architecture

Services Gatekeeper provides extensive support for authentication, authorization, and accounting. In addition, the separation of physical tiers allows for network-level security. This helps protect the network from attacks by fraudulent applications that use resources without paying for their usage and attacks designed to take resources out of service.

By using separate IP-network domains, one for the Access Tier and one for the Network Tier, you can apply different levels of network security. Applications are allowed to physically connect only to Access Tier servers, possibly fronted by a firewall, while the Network Tier servers only have access to the network domain where the telecommunication network nodes reside. In addition to this, the Access Tier servers are only allowed to connect to the Network Tier servers, possibly using a firewall between the tiers.

This topology puts the Access Tier servers in a demilitarized zone (DMZ) where out-of-network applications are only allowed access to the Access Tier servers. It also puts the Network Tier servers in a more strictly controlled domain, where the network elements they connect to are well known and the access is controlled by firewalls.

For more information, see "Deploying Services Gatekeeper in a Demilitarized Zone" in *Services Gatekeeper Security Guide*.

High Availability

From a high-availability perspective, tiered deployments are better than non-tiered deployments. Processing in the Access Tier is stateless and is characterized by negligible latency while processing in the Network Tier is stateful, involves more processing logic, and higher latency.

In a tiered deployment, the Access Tier adds a high-level load balancing function that is aware of the health of each Network Tier server and quickly removes an out of service server from the list of servers to load balance among. This means that fewer requests are affected if a fault occurs.

The Access Tier guarantees that requests toward the Network Tier are properly load balanced and are not repeatedly sent to Network Tier servers that are out of service. Network tier servers asynchronously update the Access Tier servers when they are back in service.

If a request from an Access Tier server targets a Network Tier server that has failed, the Access Tier server sends the request to another Network Tier server. The Storage Service provides reliable cluster-wide access to all state information kept by the Network Tier. As a result, any Network Tier server can process requests coming from any Access Tier node or network node. If a node fails, cluster singleton services are automatically migrated from the failed node to a healthy node.

High Availability and JavaDB

JavaDB supports one master one slave database. If the master fails, the slave completes the recovery by redoing the log that has not already been processed. The state of the slave after this recovery is close to the state the master had when it crashed. However, some of the last transactions performed on the master may not have been sent to the slave and may therefore not be reflected.

About Non-tiered Deployments

Services Gatekeeper can be deployed in a non-tiered deployment by using one of the following configurations:

- Multiple node configuration: A non-tiered, multiple node configuration is targeted toward smaller production environments that have less strict scalability and high availability requirements.
- Single node configuration: A non-tiered single node configuration is targeted toward test and development environments.

Service facades, service enablers, container services, and service interceptors are deployed in all nodes in non-tiered deployments.

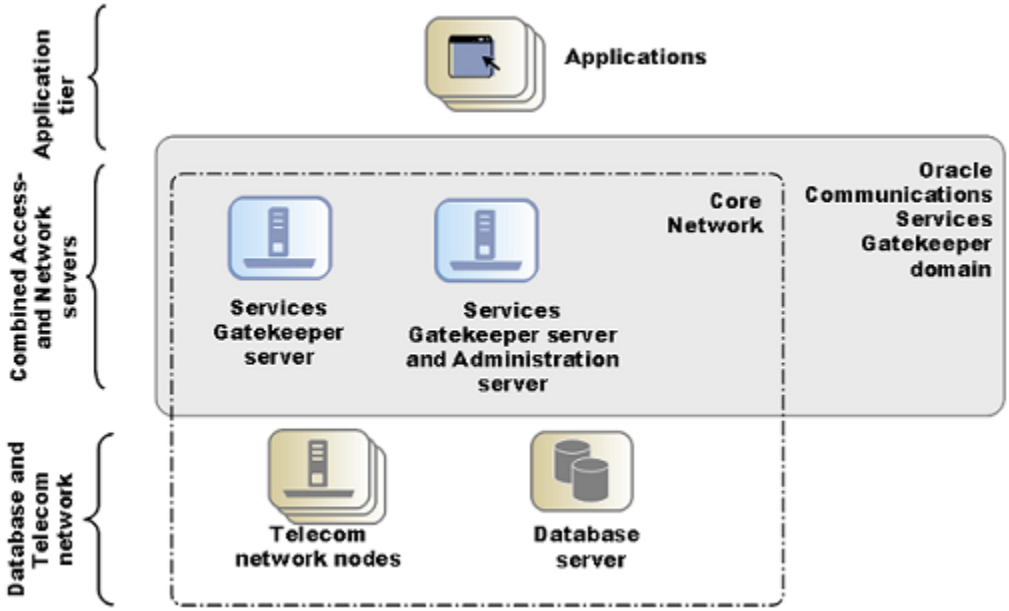
Using Non-tiered Deployments in Production Environments

You might use a non-tiered deployment in a production environment when security requirements and scalability requirements are irrelevant or minimal. An example of this is when Services Gatekeeper only serves applications hosted within the operators' domain and there is very restricted access to the IP network where Services Gatekeeper is deployed, and the integrity of the network is ensured by external mechanisms.

Scalability is compromised when you use a co-located access and Network Tier, because the individual servers cannot be optimized according to their diverse processing models.

Figure 2–5 shows a deployment that has a cluster configuration. The Administration Server also processes traffic. The cluster can contain two or more servers.

Figure 2–5 Example of a Dual Server Non-tiered Deployment for Basic High Availability

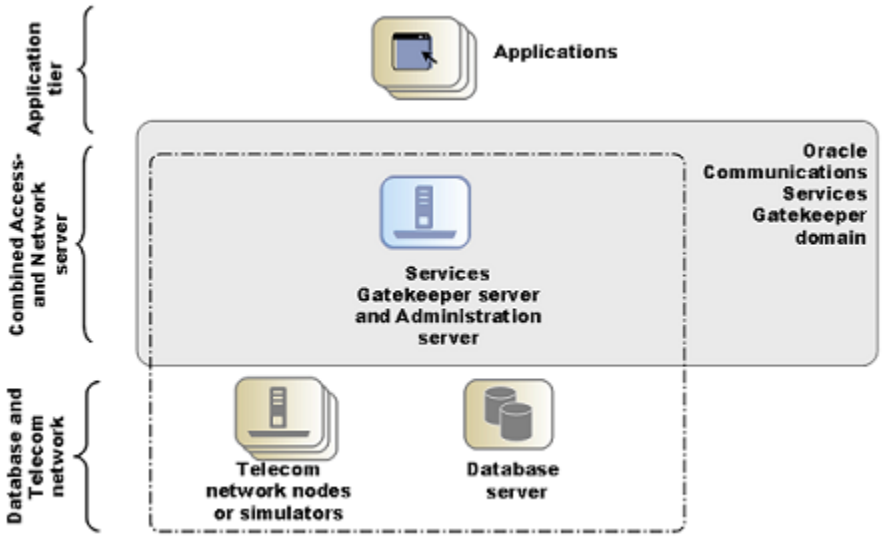


Using Non-tiered Deployments in Test and Development Environments

When you use Services Gatekeeper for functional testing of extensions and integrations, there is no immediate need for a multi-server configuration. A single-server configuration can be used to simplify management and configuration for the developer or tester.

Although it is possible to run several servers on a single physical machine, the only reason to do so is to run initial high availability tests. System tests should be performed on a deployment with multiple physical servers.

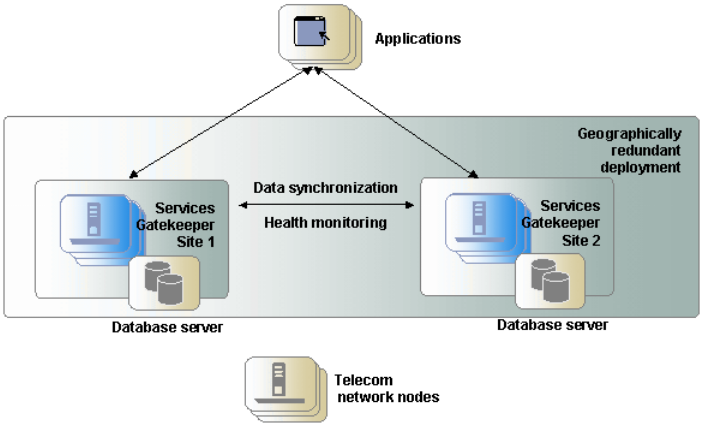
Figure 2–6 Example of a Single Node Non-tiered Deployment for Test and Development



About Geographically Redundant Deployments

Geographically separated deployments are important for high availability. To prevent service failure in the face of catastrophic events such as natural disasters or massive system outages caused by power failures, you can deploy Services Gatekeeper at two geographically distant sites that are designated as site pairs. As Figure 2–7 shows, each site, which is a Services Gatekeeper domain, has another site as its peer. Application and service provider configuration information, including related Service Level Agreements (SLAs) and budget information, is replicated and enforced across sites.

Figure 2–7 Example of a Geographically Redundant Deployment



Geographically redundant sites are the active-active type, which means that both sites in a pair can be used to process traffic simultaneously. These deployments are connected by communication channels for data synchronization and health monitoring. The data synchronization replicates budget information between the site pairs to enforce SLAs accurately. Applications should have site affinity, but have the ability to fail over to the other site if necessary. Each site is managed and configured independently. Accounts and SLAs are replicated across sites. Typically, one database tier is used per site.

All sites have a geographic site name and each site is configured to have a reference to its peer site using that name. The designated set of information is synchronized between these site peers.

One site is defined as the **geomaster**, the other as the **slave**. Checks are run periodically between the site pairs to verify data consistency and an alarm is triggered if mismatches are found, at which point the slave can be forced to synchronize to the geomaster. Any relevant configuration changes made to either site are written synchronously across the site pairs, so that a failure to write to one site causes the write to fail at the other site while triggering an alarm.

If a slave site becomes unavailable for any reason, the geomaster site becomes read-only, either until the slave site is available and has completed all data replication, or until the slave site has been removed from the geomaster site's configuration, terminating geographic redundancy. This behavior applies only to global configuration changes.

For applications, geographic redundancy means that their traffic can continue to flow in the event of a catastrophic failure at an operator site. Applications that normally use only a single site for their traffic can fail over to a peer site while maintaining ongoing SLA enforcement for their accounts. This scenario is particularly relevant for SLA aspects that have longer term impact, such as quotas.

In many respects, the geographic redundancy mechanism is not transparent to applications. There is no single sign-on mechanism across sites, and an application must establish a session with each site it intends to use. In case of site failure, an application must manually fail over to a different site.

While application and service provider budget and configuration information are maintained across sites, state information for ongoing conversations is not maintained. Conversations in this sense are defined in terms of the correlation identifiers that are returned to the applications by Services Gatekeeper or passed into Services Gatekeeper from the applications. Any state associated with a correlation identifier exists on only a single geographic site and is lost if an entire site goes down. Conversational state includes, but is not limited to, call state and registration for network-triggered notifications. This type of state is considered volatile, or transient, and is not replicated at the site level.

As a result, conversations must be conducted and completed at their site of origin. If an application wants to maintain conversational state across sites, for example, to maintain a registration for network-triggered traffic, the application must register with each site individually.

About Configuring Domains

Services Gatekeeper ships with domain configuration templates that you use to configure domains. There is a template for each type of deployment that Services Gatekeeper supports. These templates contain the basic configurations for setting up domains, but you may need to adjust some aspects of the domain during the domain configuration process.

Table 2–2 lists and explains the deployment templates.

Table 2–2 Domain Templates for Deployments

Domain Template Type	Template Name	Description
Basic developer configuration with co-located access and network tiers	**Basic Oracle Communications Services Gatekeeper Domain**	Creates an unified domain containing both the access and network tiers and the administration server all on a single machine. There is no support for high-availability configurations. The server does not belong to a cluster but is tied to the domain. This deployment type is common for non-production development machines where developers need access to Services Gatekeeper for functional testing of extensions and integrations.
Basic high-availability configuration	**OCSG Basic HA configuration**	Creates a basic high-availability, unified domain containing an access tier and network tier, each with two servers, and a database. One of the servers can also serve as the WebLogic administration server. The servers do not belong to a cluster but are tied to the domain. Database replication is not automatically provided and must be configured at the database level. This configuration can be expanded later. This deployment type is common for: ■ Non-production environments where developers need access to Services Gatekeeper for non-functional testing such as basic high-availability testing of extensions and integrations. ■ Basic, entry-level production environments that have limited requirements for security, because it does not support a DMZ separated by a firewall. It also provides minimal redundancy because it supports only two-server setups.
Access and network tier clusters	**OCSG Domain with Access and Network Clusters**	Creates a basic distributed domain, with a two-instance access tier cluster and a two-instance network tier cluster. A separate server has the role of the WebLogic administration server. This server does not process traffic requests. This configuration can be expanded later. High availability toward the database is not supported automatically. Redundancy toward the database is up to the database deployment. This deployment type is common for production environments and support deployments with a DMZ between the access and network tiers.
Access and network tier clusters with Oracle RAC database	**OCSG Domain with Access and Network Clusters with Oracle RAC Configuration**	Creates a basic distributed domain, with a two-instance access tier cluster and a two-instance network tier cluster. This configuration can be expanded later. It also creates the additional data sources required for use with an Oracle RAC-based installation. This configuration has all the properties of the access and network tier cluster setup and adds high availability and redundancy toward the database. This setup leverages the failover and redundancy characteristics of Oracle Real Application Cluster (Oracle RAC). This deployment type is common for production environments and supports deployments with a DMZ between the access and network tier.
Portal servers	**OCSG Portal Domain**	Creates a domain that contains the Portal server on a single machine.

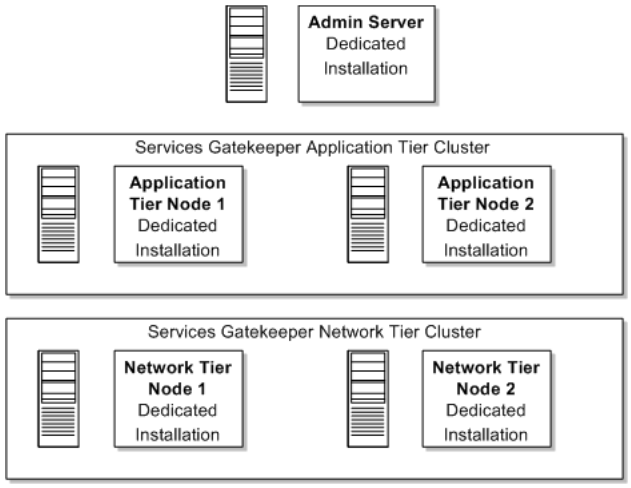
System Deployment Planning

All servers in a Services Gatekeeper cluster must be dedicated servers.

You must perform an installation on each machine in your Services Gatekeeper configuration.

Figure 2–8 shows one recommended Services Gatekeeper installation.

Figure 2–8 Recommended Services Gatekeeper Installation



For more information on more secure installations see the diagrams in "Deploying Services Gatekeeper in a Demilitarized Zone" in *Services Gatekeeper Security Guide*.

See "Installing Services Gatekeeper" for detailed installation instructions.

About Setting Up Services Gatekeeper Reporting Support

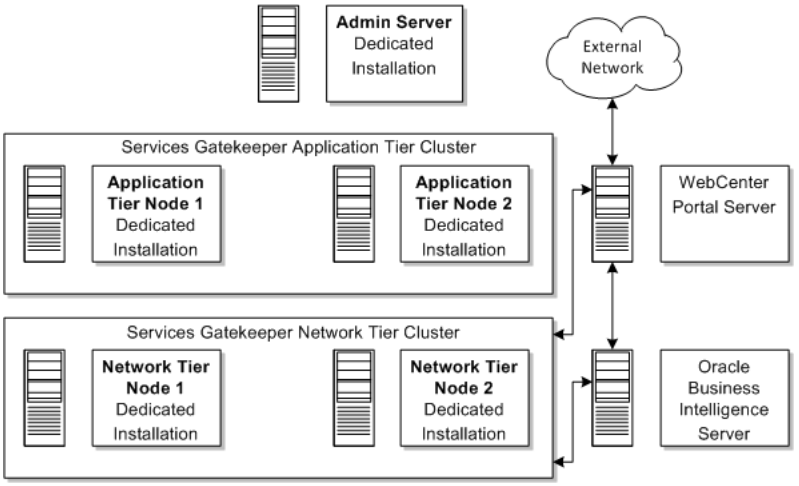
If you plan to install Services Gatekeeper reporting support, you will need additional servers to host Oracle Business Intelligence.

You install Services Gatekeeper reporting support by using a separate installer. Before you configure the reporting functionality, you must install and configure Oracle Business Intelligence, which prepares and renders the Services Gatekeeper reports.

Oracle Business Intelligence requires a dedicated server and an reports staging database.

Figure 2–9 shows a recommended Services Gatekeeper installation including a dedicated servers for Oracle Business Intelligence.

Figure 2–9 Recommended Services Gatekeeper Installation with Reporting



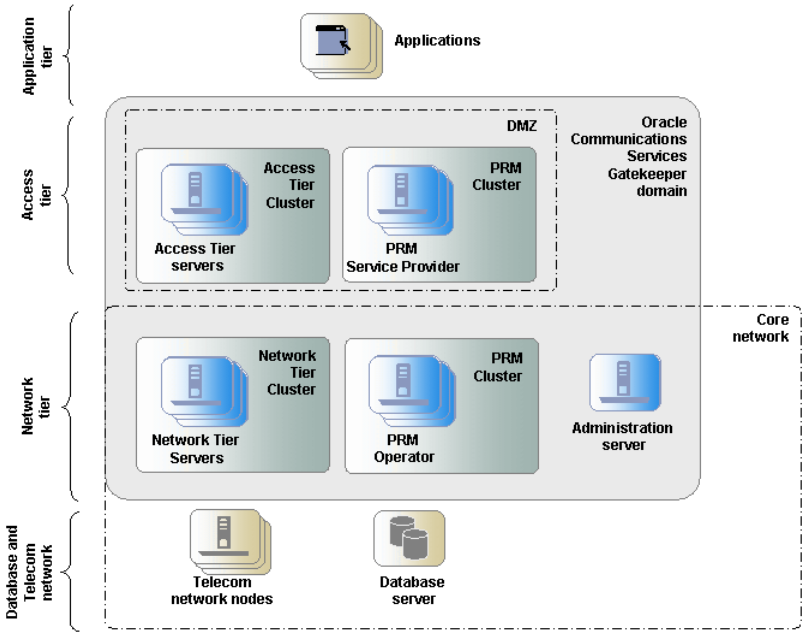
Note: A core Services Gatekeeper installation must be configured and running before you can install portal and reporting support.

About Deploying Partner Relationship Management Modules

Oracle recommends that you deploy the Partner Relationship Management (PRM) modules in a separate cluster in the domain. The PRM servers should not be co-located with Access Tier servers or Network Tier servers, because PRM processing impacts the performance of processing traffic requests.

There are two views to PRM: the service provider view and the operator view. Each view can be deployed separately. A portal application can benefit from being deployed in two parts: the service provider view deployed in the DMZ and the operator view deployed inside the secure network of the operator, not accessible from the Internet.

Figure 2–10 Example of a PRM Deployment



About Integrating Services Gatekeeper with Converged Application Server, Service Controller

You can integrate Services Gatekeeper with Oracle Communications Converged Application Server, Service Controller (Converged Application Server) if your implementation requires service orchestration and protocol mediation capabilities.

A Converged Application Server-Service Gatekeeper integration must communicate using SIP traffic, and Services Gatekeeper must then translate the SIP traffic into SS7 format. Consequently, these are the network-facing communication services that can take advantage of the integration:

- Parlay X 2.1 Audio Call/SIP
- Parlay X 2.1 Call Notification/SIP
- Parlay X 2.1 Presence/SIP
- Parlay X 2.1 Third Party Call/SIP
- RESTful Third party Call
- RESTful Call Notification
- RESTful Audio Call
- RESTful Presence

For details on these communication services see the *Services Gatekeeper Communication Service Reference Guide* and the section on RESTful services in *Services Gatekeeper Application Developer's Guide*.

About Enterprise Manager Compatibility

Services Gatekeeper is compatible with Oracle Enterprise Manager Cloud Control version 12*c*. For information about Enterprise Manager Cloud Control 12*c*, see the Enterprise Manager 12c page on the Oracle Technology Network website:

`http://www.oracle.com/technetwork/oem/enterprise-manager/overview/index.html`

See "Using Cloud Control Monitoring with Multi-tier Services Gatekeeper" in *Services Gatekeeper Administrator's Guid*e for more information.

About Administering Your Implementation

You administer Services Gatekeeper by editing the underlying MBeans, or running their operations. Services Gatekeeper provides a GUI tool for this purpose, or you can use another Java management tool. For details, see "Administration Overview" in *Services Gatekeeper System Administrator's Guide*.

Disk Storage Planning

You can use an ordinary disk system for disk storage. However, for performance and high availability reasons, a RAID system should be used.

Latency and Bandwidth Requirements

To avoid transaction processing issues related to latency or bandwidth restrictions Oracle provides the following guidelines for minimum latency and bandwidth requirements used in production environments.

Latency Requirements

Table 2–3 shows the minimum latency requirements between Services Gatekeeper entities in a production environment.

Table 2–3 Latency Guidelines for Services Gatekeeper Configurations

Configuration	Guideline
Network tier to database	Oracle recommends a latency value of less than 25 ms
Geo-redundant (site to site)	Oracle recommends a latency value of less than 1000 ms between sites

Bandwidth Requirements

Table 2–4 shows the minimum required bandwidth between Services Gatekeeper entities in a production environment. Bandwidth requirements depend on the traffic type and traffic load present in your environment. These guidelines are for typical deployments of Services Gatekeeper

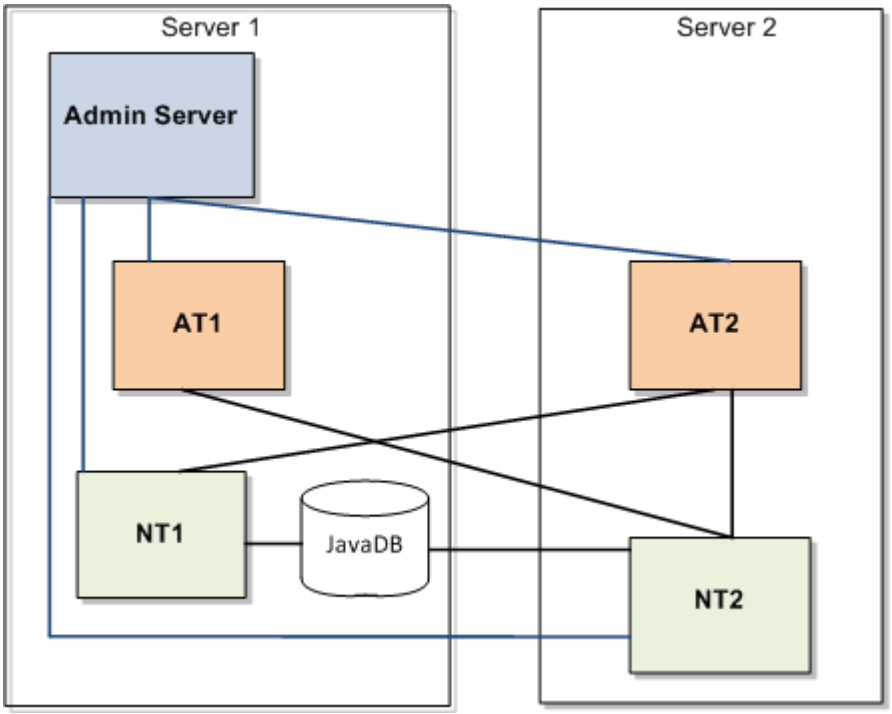
Table 2–4 Bandwidth Guidelines for Services Gatekeeper Configurations

Configuration	Guideline
Network tier to database	At least 30 Mbps/1000 tps
Application tier to network tier	Less than 15ms
Geo-redundant (site to site)	At least 1 Mbps/1000 tps

Database Planning

Production multi-tier systems generally use a database on a system separate from the other Services Gatekeeper components. Production implementation architecture strongly favors deploying a non-JavaDB database in a separate tier for performance and security reasons. However, for a test and development implementation you can deploy the JavaDB on the same system as the other Services Gatekeeper components. Figure 2–11 shows an example of this architecture that is suitable for test, or small production implementations. This architecture offers high-availability and is relatively inexpensive to implement. However, it does sacrifice security, so only use it in a trusted environment, such as for internal company API management.

Figure 2–11 Multi-tier Services Gatekeeper with a JavaDB



Installing a multi-tier Services Gatekeeper with a non-JavaDB database usually requires a different strategy. The database should be deployed on a dedicated server. Backup and other data-intensive operations should not increase load on traffic-processing servers. Database administrators should be granted exclusive privileges to log on and perform SQL operations.

Oracle recommends that you use an Oracle database, where the instance is based on the transaction processing template, runs in dedicated server mode, and uses automatic store management.

See "Supported Databases" for a list of the databases you can use.

For more information on secure production implementation, see “Deploying Services Gatekeeper in a Demilitarized Zone” in *Services Gatekeeper Security Guide*.

3

Services Gatekeeper System Requirements

This chapter summarizes the system requirements for Oracle Communications Services Gatekeeper.

Software Requirements

Table 3–1 shows the Services Gatekeeper supported software. All values in the table apply to both the Access Tier and Network Tier.

Table 3–1 Oracle Communications Services Gatekeeper Supported Platform Matrix

OS Version	OS 32/64 Bit	Processor	JDK Version	JDK 32/64 Bit
Oracle Linux 6 (UL6+)	64	x64	JDK 1.8.0_171 or later, plus the latest security updates	64
Oracle Linux 7 (UL2+)	64	x64	JDK 1.8.0_171 or later, plus the latest security updates	64
Red Hat EL 6 (UL6+)	64	x64	JDK 1.8.0_171 or later, plus the latest security updates	64
Red Hat EL 7 (UL2+)	64	x64	JDK 1.8.0_171 or later, plus the latest security updates	64
Windows Server 2016[1]	64	x64	JDK 1.8.0_171or later, plus the latest security updates	64

[1] Windows is only supported for use with development environments and is not recommended for production deployment.

Supported Databases

Table 3–2 shows the databases that Services Gatekeeper supports.

Table 3–2 Supported Databases

Database	Characteristics
Java DB 10.11.1.1 (Embedded in Services Gatekeeper in WebLogic Server 12.2.1.3.0)	Failover to one slave and fault tolerance
Oracle 12cR2 RAC	Full DB Failover and Fault Tolerance
Oracle 12*c*R2 Single Instance	No failover and fault tolerance
Oracle Database 12*c*R2 Express Edition	No failover and fault tolerance
Oracle 11*g* RAC	Full DB Failover and fault tolerance

Table 3–2 (Cont.) Supported Databases

Database	Characteristics
Oracle 11*g* Single Instance	No failover and fault tolerance
Oracle 11*g* Database Express Edition	No failover and fault tolerance
MySQL Single Instance 5.7.19	No failover or fault tolerance
MySQL Cluster 7.5.8	Full DB failover and fault tolerance

Supported Virtual Platforms

Services Gatekeeper supports the following virtual platforms:

- VMware ESXi 5.5
- Oracle Virtual Machine 3.2.9

Supported Protocols

Services Gatekeeper supports the following protocols:

- Parlay X

  Services Gatekeeper supports Parlay X Version 2.1 and Parlay X Version 3.0. For details, see the descriptions of the individual communications service in *Services Gatekeeper Communication Service Reference Guide*.

- SNMP

  Services Gatekeeper supports SNMPv1 and SNMPv2.

- IPv6

  Services Gatekeeper is fully IPv6-compliant. It provides simultaneous IPv4/IPv6 support.

About Critical Patch Updates

Install all Oracle Critical Patch Updates as soon as possible. To download Critical Patch updates, find out about security alerts, and enable email notifications about Critical Patch Updates, see the "Security" topic on the Oracle Technology Network website:

`http://www.oracle.com/technetwork/topics/security/whatsnew/index.html`

Hardware Requirements

Services Gatekeeper requires the following hardware:

- For the Services Gatekeeper software:
  - RAM: 2 GB required; 4 GB recommended
  - Disk space: 2 x 36 GB

There must be at least 1.5 GB of disk space available under **/usr/local**.

Information Requirements

Table 3–3 shows the information you must provide during the Services Gatekeeper installation.

Table 3–3 Required Information

Information Type	Description	Default Value
Host name or IP address	The network names or IP addresses of the machines on which you are going to install the software.	Current host name or IP address
Directory	The directory on each machine that will serve as your *Middleware_home* directory.	NA
Directory	The directories on each machine in which to install Services Gatekeeper and WebLogic Server software. By default, these are subdirectories of *Middleware_home*.	NA
Directory	If you are going to install Services Gatekeeper Platform Development Studio, the Eclipse plug-in directory.	NA
Password	A password for the administrative user. The password must have a minimum of eight characters, at least one of which is non-alphabetic.	NA

4

Installing the Database

This chapter provides an overview of installing a supported database for use with Oracle Communications Services Gatekeeper.

Database Installation Overview

There are substantial differences between the installation procedures for the different supported databases. JavaDB is included with the Services Gatekeeper software, so it is the only database that you not need to obtain separately, and install first. The other database installations follow these basic steps:

1. Installing the database software.
2. Setting up a user account that Services Gatekeeper uses to access the database.
3. Granting the user account appropriate privileges on the database.

See the following sections for information about the database you are installing:

- Installing JavaDB Software
- Installing Oracle RAC or Oracle Single Instance Database Software
- Installing Oracle Database Express Edition
- Installing MySQL Database
- Installing MySQL Cluster CGE

Installing JavaDB Software

Follow the instructions in this section to install JavaDB for Services Gatekeeper to use. JavaDB is included in the Services Gatekeeper software, so this is the only section in this chapter that instructs you to install Services Gatekeeper first.

About Using JavaDB with Services Gatekeeper

JavaDB is appropriate for test and development implementations and smaller production environments. The instructions below direct you to install the Services Gatekeeper on two physical servers. The first server includes the administration server, along with the AT and NT managed servers, and the JavaDB. The second server includes additional AT and NT servers. Colocating the JavaDB with the other Services Gatekeeper components sacrifices security, but it also achieve high-availability, and is inexpensive. For production environments, install it on a separate physical system so that it is more secure.

If just you want a single-tier implementation, see the instructions in "Getting Started with Services Gatekeeper" in *Services Gatekeeper Getting Started* to install everything on one system.

Installing the JavaDB Software

To install the JavaDB software:

1. Install the Services Gatekeeper administration server and an NT server and AT server on one physical system. Record the IP address and a listening port number.

   See "Installing Services Gatekeeper" for details.

2. On the system you just installed, navigate to *Oracle_home*/**wlserver/common/derby/bin**.

3. Start the DB service with this command:

   ```
   ./startNetworkServer.sh -h NT1_IP_addr
   ```

   Where *NT1_IP_addr* is the IP address of the NT server system you installed with Services Gatekeeper.

4. Navigate to *Oracle_home*/**wlserver/common/derby/lib.**

5. Start the JavaDB tool and create the database with this command:

   ```
   java -jar derbyrun.jar ij
   ij> connect 'jdbc:derby://NT1_IP_addr:port_no/RemoteDb;create=true';
   ```

   Where *NT1_IP_addr* is the IP address of the NT server system you installed with Services Gatekeeper, and *port* is the listening port to use.

6. Create a database user with this command:

   ```
   ij> derby.user.username=password
   ```

   Where *username* and *password* are the name and password of the database administrator.

7. Create the DUAL database table in JavaDB with this command:

   ```
   ij> connect 'jdbc:derby://NT1_IP_addr:port_no/RemoteDb';
   ij> create table DUAL(num int);
   ij> disconnect
   ```

   Where *NT1_IP_addr* is the IP address of the NT server system you installed with Services Gatekeeper, and *port* is the listening port to use.

8. Perform any required post installation tasks

   See "Services Gatekeeper Post-Installation Tasks" for information.S

9. Configure your domains.

   See "Configuring the Services Gatekeeper Domain" for information.

10. Install a second server with the Services Gatekeeper NT and AT managed servers. See "Installing Services Gatekeeper" for instructions. We will refer to this system as server2. Record the IP address and the Services Gatekeeper username.

11. Run this command on the server that contains the Services Gatekeeper administration server and JavaDB:

    ```
    scp -r user_projects/ username@server2_IP_addr:install_dir
    ```

Where *username* is the name of the Services Gatekeeper administrator on Server2, *server1_IP_addr* is the IP address of Server2, and *install_dir* is the root directory of the installation.

You can use the `ij` SQL scripting tool or a third-party database administration tool such as SQuirrel SQL Client to administer the database.

Installing Oracle RAC or Oracle Single Instance Database Software

Follow these instructions if you are using Oracle Real Application Clusters (Oracle RAC) or Oracle Database Single Instance database software.

The database must be installed on a dedicated server running outside the Services Gatekeeper cluster.

About Using Oracle RAC with WebLogic Server

Oracle RAC or MySQL are supported production environments that require high availability.

For information about using Oracle WebLogic Server with multiple data sources, see *Oracle Fusion Middleware Administering JDBC Data Sources for Oracle WebLogic Server*.

Installing the Database Software

To install the database software:

1. Follow the instructions in the database installation guide, available in the installing and upgrading section on the *Oracle Database Documentation Library* website.

   During the installation process, select these configuration options:

   - Create the database using the **Transaction Processing** template.
   - Use the **Dedicated Server Mode** for the database.
   - Change the **processes** parameter to be equal to:

     [(`wlng.datasource MaximumCapacity + wlng.localTX.datasource MaximumCapacity`) * *NumberOfServers*]

     where *NumberOfServers* is the number of Services Gatekeeper servers in the cluster. **MaximumCapacity** is a parameter in the connection pool settings for the JDBC data sources. Normally this value is 150 for both data sources, but you may need to increase it.

2. Download and install the latest Oracle database patch set.
3. Continue to "Setting Up a Services Gatekeeper User for the Oracle Database".

Setting Up a Services Gatekeeper User for the Oracle Database

To set up a Services Gatekeeper user for the Oracle database:

1. Create a database user for Services Gatekeeper with an allowed (unlimited) quota on its default tablespace (the **users** tablespace). The user name and password for the user are later copied to each Services Gatekeeper server.
2. Grant the user the following privileges:
   - CREATE SESSION

- CREATE TABLE

3. Continue to "Installing Services Gatekeeper".

Installing Oracle Database Express Edition

Follow the instructions in this section if you are using Oracle Express Edition (XE) as your database.

Oracle XE can be installed either on a server in the Services Gatekeeper cluster or on a separate server. If it is installed in the cluster, it should be in the same server as the Network Tier.

Note: Oracle XE is recommended over MySQL for Services Gatekeeper development installations because the Oracle XE schema is compatible with enterprise Oracle databases. Neither Oracle XE nor MySQL is recommended for production deployment.

Installing the Oracle XE Software

To install Oracle XE:

1. Download the Oracle Database Express Edition installer from the Oracle Technology Network website:

   `http://www.oracle.com/technetwork/index.html`

2. Follow the instructions to select and download the Oracle Database Express Edition software for your operating system.
3. Install Oracle XE using the instructions in the installation guide on the Oracle documentation website at:

   `https://docs.oracle.com/en/database/`

4. Continue to "Configuring Oracle XE for Services Gatekeeper".

Configuring Oracle XE for Services Gatekeeper

To configure Oracle XE for Services Gatekeeper:

1. Open a command window.
2. (Linux only) If the required environment variables are not already set, do the following:

   For Bash, Bourne, or Korn shell, enter the following command:

   ```
   source ORACLE_HOME/bin/oracle_env.sh
   ```

   For C shell, enter the following command:

   ```
   source ORACLE_HOME/bin/oracle_env.csh
   ```

3. Enter the following command:

   ```
   sqlplus /nolog
   ```

4. Connect to the database (on Windows you are prompted for the user name and password):

   ```
   SQL> connect SYSTEM/SYSTEM_user_password@XE
   ```

5. Enter the following command to increase the number of allowable JDBC connections:

   ```
   SQL> alter system set processes=300 scope=spfile;
   ```

6. Create a Services Gatekeeper user and password using the following command:

   ```
   SQL> create user database_username identified by password;
   ```

7. Grant the newly created user privileges using the following command:

   ```
   SQL> grant create session, create table, resource to database_username;
   ```

8. Exit SQL*Plus:

   ```
   SQL> exit
   ```

9. Restart the database for the changes to take effect.
10. Continue to "Installing Services Gatekeeper".

Installing MySQL Database

Follow the instructions in this section if you are using MySQL as your database. Services Gatekeeper supports using single instance and clustered MySQL database implementations.

MySQL can be installed either on a server in the Services Gatekeeper cluster or on a separate server. If it is installed in the cluster, it should be in the same server as the Network Tier.

> **Note:** Oracle XE is preferable to MySQL for Services Gatekeeper development installations because the Oracle XE schema is compatible with enterprise Oracle databases. Oracle XE is not recommended for production deployment.

This section covers the following topics:

- Installing the MySQL Database Software
- Configuring MySQL on Linux
- Configuring MySQL on Windows
- Creating the Database and a Database User

Installing the MySQL Database Software

To install a MySQL database:

1. Download the MySQL database software from the Oracle software delivery website at:

   https://edelivery.oracle.com

2. Follow the instructions in the MySQL documentation for installing MySQL. The documentation is available on the MySQL website at:

   http://dev.mysql.com

3. When installing MySQL:

- **Linux**: For most Linux distributions, you can use a package manager such as **dpkg** or YUM.
- **Windows**: Unless you have additional special requirements, you can select the **Developer Default** installation option.

4. Continue to one of the following sections:
   - Configuring MySQL on Linux
   - Configuring MySQL on Windows

Configuring MySQL on Linux

This section summarizes the commands required to configure MySQL on most versions of Linux. Command locations may differ between Linux distributions.

To configure MySQL on Linux:

1. As the user root, start the MySQL database:

   ```
   /etc/rc.d/init.d/mysqld start
   ```

2. Open the **/etc/my.cnf** file and do the following:
   - Edit the connection variable so that **max_connections** is equal to:

     [(`wlng.datasource MaximumCapacity + wlng.localTX.datasource MaximumCapacity`) * *NumberOfServers*]

     where *NumberOfServers* is the number of Services Gatekeeper servers in the cluster. **MaximumCapacity** is a parameter in the connection pool settings for the JDBC data sources. Normally this value is 150 for both data sources, but you may need to increase it.

     ```
     [mysqld]
     max_connections=400
     ```

   - Add an entry for the default character set. The recommended character set is **Latin1**.

     ```
     default-character-set=latin1
     ```

3. Save and close the file.
4. Restart MySQL:

   ```
   /etc/rc.d/init.d/mysqld stop
   /etc/rc.d/init.d/mysqld start
   ```

5. Continue to "Creating the Database and a Database User".

Configuring MySQL on Windows

To configure MySQL on Windows:

1. From a text editor, open the **my.ini** file and do the following:
   - Edit the connection variable so that **max_connections** is equal to:

     [(`wlng.datasource MaximumCapacity + wlng.localTX.datasource MaximumCapacity`) * *NumberOfServers*]

     where *NumberOfServers* is the number of Services Gatekeeper servers in the cluster. **MaximumCapacity** is a parameter in the connection pool settings for

the JDBC data sources. Normally this value is 150 for both data sources, but you may need to increase it.

```
[mysqld]
max_connections=400
```

- Add an entry for the default character set. The recommended character set is **Latin1**.

```
default-character-set=latin1
```

2. Save and close the file.
3. Continue to "Creating the Database and a Database User".

Creating the Database and a Database User

To configure MySQL for Services Gatekeeper, perform the following for each IP address in the cluster:

1. Create the Services Gatekeeper database user and password with this command:

```
CREATE USER database_username@ip_address IDENTIFIED BY user_password
```

You will need to provide this user name and password when you configure the Services Gatekeeper domain.

For information about the various command-level modes of accessing the MySQL server, see the documentation on the MySQL website.

2. Grant access privileges:

```
GRANT ALL ON *.* TO database_username@ip_address IDENTIFIED BY user_password
```

3. Create the database for Services Gatekeeper:

```
CREATE DATABASE database_name
```

You will need to provide the database name when you configure the Services Gatekeeper domain.

4. Continue to "Installing Services Gatekeeper".

Installing MySQL Cluster CGE

Follow the instructions in this section if you are using MySQL Cluster Carrier Grade Edition (CGE) as your database. Services Gatekeeper supports using single instance and clustered MySQL database implementations.

MySQL can be installed either on a server in the Services Gatekeeper cluster or on a separate server. If it is installed in the cluster, it should be in the same server as the Network Tier.

Installing the MySQL Cluster CGE Software

To install MySQL Cluster SGE software:

1. Download a supported version of MySQL Cluster from the Oracle software delivery website:

https://edelivery.oracle.com/

2. Install the MySQL Cluster CGE software using the instructions in the installation guide on the Oracle documentation website:

   http://docs.oracle.com/cd/E17952_01/index.html

Configuring the Management Server Node

For a first Cluster, start with a single MySQL Server (mysqld), a pair of Data Nodes (ndbd) and a single management node(ndb_mgmd) – all running on the same server. Using the management server node, you can start and stop other nodes, configure data, run backup, and perform other tasks.

To configure the management server node:

1. Create the **config.ini** file on the management server, under the **/var/lib/mysql-cluster/** directory.

   The following is an example of the **ndbd default** section in the **config.ini** file. For the **hostname** parameter, *host_name_or_IP_address* represents the host name or IP address of the node. For example:

   hostname=node1.example.com

   The system values in this example are suggested values. The values you use will depend on how your system is set up.

   Example **ndbd default** section of the **config.ini** file:

```
[ndbd default]
NoOfReplicas=2
DataMemory=4G
IndexMemory=400M
MaxNoOfAttributes=500000
MaxNoOfTables=1760
MaxNoOfOrderedIndexes=3000
MaxNoOfUniqueHashIndexes=1250
MaxNoOfConcurrentOperations=100000
[ndb_mgmd]
NodeId=1
hostname=host_name_or_IP_address
datadir=/var/lib/mysql-cluster/
[ndbd]
NodeId=2
hostname=host_name_or_IP_address
datadir=/usr/local/mysql/data/
[ndbd]
NodeId=3
hostname=host_name_or_IP_address
datadir=/usr/local/mysql/data/
[mysqld]
NodeId=4
hostname=host_name_or_IP_address
[mysqld]
NodeId=5
hostname=host_name_or_IP_address
[mysqld]
[mysqld]
```

2. Create the **my.cnf** file under the **/etc** directory, using the following values:

```
[mysqld]
ndbcluster
```

```
datadir=/usr/local/mysql/data
basedir=/usr/local/mysql
user = mysql
port = 3306
default-storage-engine=ndb
ndb-connectstring= host_name_or_IP_address
[mysql_cluster]
ndb-connectstring= host_name_or_IP_address
```

Starting the MySQL Cluster Processes

To start the MySQL Cluster processes:

1. Enter the following commands in order:

   ```
   ./ndb_mgmd -f /var/lib/mysql-cluster/config.ini
   ./ndbd
   mysqld_safe--ndb_nodeid=4 --user=mysql&
   ```

   After you run these commands, the **ndb_mgm** prompt appears.

2. Check the status of the cluster by entering the following command:

   ```
   ndb_mgm> show
   ```

   The status of the cluster is displayed on the command line. For example:

   ```
   Connected to Management Server at: 12.345.67.81:1186
   Cluster Configuration
   ---------------------
   [ndbd(NDB)]     2 node(s)
   id=2    @12.345.67.82  (mysql-5.5.30 ndb-7.2.12, Nodegroup: 0, Master)
   id=3    @12.345.67.83  (mysql-5.5.30 ndb-7.2.12, Nodegroup: 1)

   [ndb_mgmd(MGM)] 1 node(s)
   id=1    @12.345.67.82  (mysql-5.5.30 ndb-7.2.12)

   [mysqld(API)]   3 node(s)
   id=4    @12.345.67.82  (mysql-5.5.30 ndb-7.2.12)
   id=5    @12.345.67.83  (mysql-5.5.30 ndb-7.2.12)
   id=6 (not connected, accepting connect from any host)
   ```

   Wait for the data nodes to finish starting.

3. Start your MySQL server.

5 Installing Services Gatekeeper

This chapter describes how to install Oracle Communications Services Gatekeeper.

Before installing Services Gatekeeper, read these chapters:

- Services Gatekeeper Installation Overview
- Planning Your Services Gatekeeper Installation
- Services Gatekeeper System Requirements

About Installing Services Gatekeeper

You must install Services Gatekeeper on every server in your implementation. You must install the software in a directory that resides on the server's local file system.

You install Services Gatekeeper by using a generic installer that works on any supported operating system and for both 64-bit and 32-bit platforms.

Installation Prerequisites

The generic installer does not include a bundled JDK. The JDK must already be installed when you use the generic installer.

Installing the JDK and JCE

You must download and install a supported JDK on the target system before installing Services Gatekeeper. See "Installing the JDK and JCE" in *Services Gatekeeper Getting Started Guide* for instructions.

Creating an Installation Log

To create an installation log, add the following option to any of the commands that launch the installer:

-log=*logfilename*

where *logfilename* is a name that you assign to the log file.

For example, the following command runs the installer in GUI mode and creates a log file named **install_log** containing the installation's output.

```
java -jar ocsg_multitier_generic.jar -log=install_log
```

Installing Services Gatekeeper in GUI Mode

This section describes how to install Services Gatekeeper in GUI mode.

Important:

- Installing Services Gatekeeper on Windows is supported only for development and test environments; it is not supported for production.
- If you are installing on Windows, you must log in as an administrator.
- After starting the installer, you can cancel the installation at any time by clicking **Exit**.

To install Services Gatekeeper in GUI mode:

1. Log in to the target system.
2. Download the Services Gatekeeper installer from the Oracle software delivery website:

   https://edelivery.oracle.com/
3. Change to the directory where you downloaded the software.
4. Start the installer.
   - To start the installer on a system that uses a 32/64-bit hybrid JDK, enter:

     ```
     java -d64 -jar ocsg_multitier_generic.jar [-log=logfilename]
     ```
   - To start the installer on a 32-bit system, enter:

     ```
     java -jar ocsg_multitier_generic.jar [-log=logfilename]
     ```

   After the installer starts, the Welcome screen appears.
5. Click **Next**.

   The Installation Location screen appears.
6. In the **Oracle Home** field, enter the full path to the directory of your *Middleware_home* or use the **Browse** button to locate the directory.

   The *Middleware_home* directory is the central directory for all Oracle products installed on the target system.

   To see a list of Services Gatekeeper products that are currently installed in the directory, click **View**.
7. Click **Next**.

   The Installation Type screen appears.
8. Specify the components to install on this system by doing one of the following:
   - To install only the Administration Server, select **Administration Server** and click **Next**.
   - To install only the Network Tier, select **Network Tier** and click **Next**.
   - To install only the Portal server:
     a. Select **Portal** and click **Next**.

The Portal Parameters screen appears.

b. In the **Access Tier Servers** field, enter the address for each Access Tier server in your Services Gatekeeper implementation. The address uses the format *IPAddress*:*Port*. Each address must appear on its own line.

The IP address and port of the backend server must match the IP address and port of the Access Tier. To view the Access Tier server port, in the Administration Console, click **Servers**, then **AT Server Name**, and then **Listen Port**.

c. Click **Next**.

- To install only the Access Tier, select **Access Tier** and click **Next**.
- To install multiple Services Gatekeeper components:

a. Select **Custom Installation** and click **Next**.

The Features to Install screen appears.

b. Select the checkbox for each component that you want to install on this system. To install all of the components, select the **Oracle Communication Services Gatekeeper** checkbox.

c. Click **Next**.

The Prerequisite Checks screen appears.

9. The screen automatically tests your system to ensure that it meets all operating system and JDK software requirements:
   - A green check mark indicates that your system passed the prerequisite check.
   - A red circle indicates a problem. The bottom of the screen shows a short error message to help you troubleshoot the problem. Fix the error and click **Rerun** to perform the prerequisite checks again. To continue the installation without fixing the problem, click **Skip**.
10. Click **Next**.

    The Installation Summary screen appears.
11. Ensure that the listed installation location and feature sets to install are correct.

    If the list is not correct, you can use the **Back** button to make corrections.

    To save the information to a response file so you can install the component later, click **Save Response File** and specify the name and location of the response file.
12. Click **Install** to start the installation.

    The Installation Progress screen appears, and a progress bar indicates the status of the installation process.
13. Click **Next**.
14. When the Installation Complete screen appears, do one of the following:

Important: If you plan to use Services Gatekeeper with an IPv6 network, do not configure the domain now. You must perform certain post-installation tasks first.

- To configure your Services Gatekeeper domain now, click **Finish**.

The WebLogic Server Configuration Wizard starts.

- To complete installation without configuring your domain, deselect **Automatically Launch the Configuration Wizard** and click **Finish**.

  The Services Gatekeeper installer exits.

After installation completes, the WebLogic Server Configuration Wizard starts by default. For information about how to configure your Services Gatekeeper domain, see "Configuring the Domain in GUI Mode".

Installing Services Gatekeeper in Silent Mode

Silent mode is a way of setting installation options once and then using those settings to duplicate the installation on many machines. The installation program reads your settings from a Response File that you create prior to beginning the installation. The installation program does not display any options during the installation process. Silent-mode installation works on Windows, Solaris, and Linux systems.

About the Response File

The entries in the Response File (**response.rsp**) correspond to the prompts that you would see if you used GUI mode.

Incorrect entries in the Response File can cause the installation to fail. To help you determine the cause of a failure, Oracle recommends that you create a log file when you start the installation.

The following is a sample version of the Response File. Your input may be slightly different, depending on your installation.

```
[ENGINE]

#DO NOT CHANGE THIS.
Response File Version=1.0.0.0.0

[GENERIC]

#The Oracle home location. This can be an existing Oracle Home or a new Oracle
Home.
ORACLE_HOME=c:\oracle\gatekeeper

#Set this variable value to the Installation Type selected (for example, WebLogic
Server, Coherence, Complete with Examples).
INSTALL_TYPE=Complete with Examples

#Provide the My Oracle Support Username. If you want to ignore Oracle
Configuration Manager configuration, provide an empty string
#for the user name.
MYORACLESUPPORT_USERNAME=

#Provide the My Oracle Support Password
MYORACLESUPPORT_PASSWORD=<support_password>

#Set this to true if you want to decline the security updates. Setting this to
true and providing an empty string for the My Oracle Support
#username will ignore the Oracle Configuration Manager configuration.
DECLINE_SECURITY_UPDATES=true

#Set this to true if My Oracle Support Password is specified.
```

```
SECURITY_UPDATES_VIA_MYORACLESUPPORT=false

#Provide the Proxy Host.
PROXY_HOST=

#Provide the Proxy Port.
PROXY_PORT=

#Provide the Proxy Username.
PROXY_USER=

#Provide the Proxy Password.
PROXY_PWD=<SECURE VALUE>

#Type String (URL format) Indicates the OCM Repeater URL which should be of the
format [scheme[Http/Https]]://[repeater host]:[repeater port]
COLLECTOR_SUPPORTHUB_URL=
```

Returning Exit Codes to the Console

When run in silent mode, the installation program generates exit codes that indicate the success or failure of the installation. Table 5–1 describes these exit codes:

Table 5–1 Installation Program Exit Codes

Code	Description
0	Installation completed successfully.
-1	Installation failed due to a fatal error.
-2	Installation failed due to an internal XML parsing error.

When you start the silent-mode installation process from a script, you can use the **echo** command to have these exit codes displayed to the console. The following is a sample command file that invokes the installer in silent mode and sends the exit codes to the console.

Example 5–1 Script that Returns Exit Codes

```
rem Execute the installer in silent mode
@echo off
java -jar ocsg_multitier_generic.jar -silent
-responseFile=/home/use/bin/response.rsp -log=logfilename

@rem Return an exit code to indicate success or failure of installation
set exit_code=%ERRORLEVEL%

@echo.
@echo Exitcode=%exit_code%
@echo.
@echo Exit Code Key
@echo ---------------
@echo  0=Installation completed successfully
@echo -1=Installation failed due to a fatal error
@echo -2=Installation failed due to an internal XML parsing error
@echo.
```

Running the Installer in Silent Mode

To install Services Gatekeeper in silent mode on any supported platform:

1. Log in to the target system.
2. Download the Services Gatekeeper installer from the Oracle software delivery website:

   `https://edelivery.oracle.com/`
3. Create a **response.rsp** file, as described in "About the Response File".
4. Change to the directory where you downloaded the software.
5. Start the installer by entering the following command:

   ```
   java -jar ocsg_multitier_generic.jar -silent -responseFile ResponseFile
   ```

   where *ResponseFile* is the full path and name of the Response File. For example, **/home/use/bin/response.rsp**.
6. Check whether the installer completed successfully by retrieving the exit codes, as described in "Returning Exit Codes to the Console".

Where to Go from Here

If you want to install:

- Services Gatekeeper Reports, see "Installing Services Gatekeeper Reports".
- Services Gatekeeper Platform Test Environment (PTE), see "Installing the Platform Test Environment".

Otherwise, perform the tasks in "Services Gatekeeper Post-Installation Tasks".

6

Services Gatekeeper Post-Installation Tasks

This chapter provides instructions for Oracle Communications Services Gatekeeper post-installation tasks. You must install Services Gatekeeper before following these procedures. See "Installing Services Gatekeeper".

Overview of Services Gatekeeper Post-Installation Tasks

After installing Services Gatekeeper, you perform the following tasks:

1. Set the WebLogic Server home path.
2. Configure the Services Gatekeeper domain.
3. Perform post-installation tasks for the Services Gatekeeper installation.
4. Perform post-installation tasks for any optional components that you installed, which may include:
   - Services Gatekeeper Reports
   - Services Gatekeeper Platform Test Environment
   - Services Gatekeeper Application Test Environment

Setting the WebLogic Server Home Path

To set the WebLogic Server home path:

1. Set the WL_HOME variable to the directory in which you installed the WebLogic Server software. For example:

   ```
   WL_HOME=Middleware_home/wlserver
   ```

2. Export WL_HOME. For example:

   ```
   export WL_HOME
   ```

Configuring Your Services Gatekeeper Domain

In order to run Services Gatekeeper, its container (Oracle WebLogic Server) must be given basic information about the various parts of the system. This is called configuring the domain.

You configure the domain by running the WebLogic Server Configuration Wizard or by using the WebLogic Scripting Tool (WLST). For instructions, see "Configuring the Services Gatekeeper Domain".

Post-Installation Tasks for Services Gatekeeper

Perform these tasks on systems where you installed Services Gatekeeper.

Creating JMS Servers for Additional Network Tier Servers

If you added Network Tier servers in addition to the initial two provided by the default clustered domain templates, you must configure Services Gatekeeper to add support for the EDR Service on each server. Each server in the Network Tier requires its own JMS server in order for the EDR Service to work correctly.

For the following task, you must start the administrative server in your Services Gatekeeper installation so that you can use the Administration Console to make the necessary adjustments. Unless you are setting up an all in one domain, you also need to start at least one Network Tier server (this prevents a null pointer error when initializing the Administration Console). For more information about using the Administration Console, see *Services Gatekeeper System Administrator's Guide*.

To create the required JMS servers:

1. Start the Administration Server.
2. In a command window, go to the **domain/bin** directory.

   In the default installation, this would be *Middleware_home***/user_projects/domains/base-domain/bin**.
3. Run the following command:
   - Linux/Solaris:

     ```
     sh startWebLogic.sh
     ```
   - Windows:

     ```
     startWebLogic.cmd
     ```

   The Administration Server starts and displays output in the command window. Wait until the prompt indicates that the server is in **RUNNING** state.

   Note: This script works best with the Bash shell. If the server fails to start and returns this error:

   ```
   ./dbController.sh: 3: -/dbController.sh: Syntax Error: "(" unexpected
   ```

   edit the **startWeblogic.sh** script, changing the **#!/bin/sh** shebang to **#!/bin/bash**.

   ---
4. If you are setting up an all-in-one domain, skip this step. Otherwise, do the following:
   a. (Solaris only) Add the following line to the *Middleware_home***/user_projects/domains/base-domain/bin/startManagedWebLogic.sh** script:

      ```
      JAVA_OPTIONS="${JAVA_OPTIONS} -Dweblogic.ThreadPoolSize=100
      -Dweblogic.ThreadPoolPercentSocketReaders=50"
      ```
   b. Save and close the file.
   c. Start a Network Tier server by doing one of the following:

- **Run the start script from the network tier server:** Log in to the Network Tier server and run the **startManagedWebLogic.sh** script.
- **Run the start script from the domain/bin directory:** In a separate command window, go to *Middleware_home*/**user_projects/domains/base-domain/bin** and enter the following command:

  ```
  sh startManagedWebLogic.sh network_node t3://admin_host:port
  ```

  where *network_node* is the name of the Network Tier server, and *admin_host* and *port* are the host name and port number of the Administration Server.

Watch the command window as the Network Tier server loads. Wait until the prompt indicates that the server is in **RUNNING** state.

5. When both servers are in **RUNNING** state, start the Administration Console.

   In your browser, enter the following address:

   ```
   http://hostname:port/console
   ```

   where *hostname* is the host name of the Administration Server, and *port* is the port number used for the listen address assigned during domain configuration.

6. Log in using your login credentials.

   If this is the first time you have logged in, you should use username: `weblogic` and a password that you create. There are instructions in *Services Gatekeeper System Administrator's Guide* on changing these values after your system is fully configured.

7. Click **Lock & Edit** in **Change Center**.

8. Create the new JMS server:

   a. In the Administration Console, select **Home**, then **Services**, then **Messaging**, and then **JMS Servers**.

   b. Click **New**.

   c. In the **Name** field, enter the name of the JMS Server.

   d. From the **Target** menu, select the Network Tier server on which to create the JMS server.

   e. Click **Finish**.

   f. Click **Activate Changes**.

(Optional) Adding a Custom Password Validator

You can add a custom password validator to Services Gatekeeper by using features available through Oracle WebLogic Server. To do so, you create and configure a **Password Validation Provider**. This allows you to enforce rules concerning the composition of passwords used with Services Gatekeeper. In general, the rules include:

- Whether the password may contain the user's name, or the reverse of that name
- A minimum or maximum password length (composition rules may specify both a minimum and maximum length)
- Whether and how many of the following characters must be in the password:
  - Numeric characters
  - Lowercase alphabetic characters

- Uppercase alphabetic characters
- Non-alphanumeric characters (for example, parentheses or asterisks)

For more information about adding password validation to your Services Gatekeeper installation, see "Configuring the Password Validation Provider" in *Oracle Fusion Middleware Administering Security for Oracle WebLogic Server*.

(Optional) Adding Java Cryptography Extensions

Services Gatekeeper does not require Java Cryptography Extensions (JCE) features to run, but you can install them if your implementation requires them. For more information about adding JCE, see "Using JCE Providers with WebLogic Server" in *Oracle Fusion Middleware Administering Security for Oracle WebLogic Server*.

Post-Installation Tasks for Reports

Perform these tasks if you installed Services Gatekeeper Reports as described in "Installing Services Gatekeeper Reports".

Configuring the Reports Data Source

Before you begin, make sure you have the following information for your reports database.

- Database Name
- Host Name
- Database Server Port
- Database User Name
- Database User's Password

To configure the reports staging data source:

1. Make sure that the Services Gatekeeper Administration Server is running.
2. Start the Administration Console by entering the following URL in your web browser:

   `http://hostname:port/console`

   Where *hostname* is the DNS name or IP address of the Services Gatekeeper Administration Server and *port* is the address of the port on which the Administration Server is listening for requests (8001 by default).
3. When the login page appears, enter the user name and the password you used to start the Administration Server (you may have specified this user name and password during the Services Gatekeeper installation process), or enter a user name that has been granted one of the default global security roles.
4. In the Change Center of the Administration Console, click **Lock & Edit**.
5. In the **Domain Structure** tree, select your Services Gatekeeper domain and expand **Services**, then **JDBC**, and then select **Data Sources**.
6. On the Summary of Data Sources page, click **New** and choose **Generic Data Source** from the list.
7. On the JDBC Data Sources Properties page, specify the following information:

- **Name**: Enter the following name for the JDBC data source: **analytic.datasource**
- **JNDI Name**: Enter the following path to the JDBC data source: **oracle.ocsg.edr.analytic**
- **Database Type**: Select the DBMS type of the database you're using as your reports staging database. If your DBMS is not listed, select Other.

Click **Next** to continue.

8. Select the JDBC driver you want to use to connect to the database.

   Note: You must install JDBC drivers before you can use them to create database connections. Some JDBC drivers are installed with WebLogic Server, but many are not installed.

   Click **Next** to continue.

9. On the Connection Properties page, enter values for the following properties:
   - **Database Name**: Enter the name of your reports database.
   - **Host Name**: Enter the DNS name or IP address of the server hosting the reports database.
   - **Port**: Enter the port on which the database server listens for connections requests.
   - **Database User Name**: Enter the reports database username.
   - **Password/Confirm Password**: Enter the password for the reports database user.

   Click **Next** to continue.

10. On the Test Database Connection page, review the connection parameters and click **Test Configuration**.

    Services Gatekeeper attempts to create a connection from the Administration Server to the database. Results from the connection test are displayed at the top of the page. If the test is unsuccessful, you should correct any configuration errors and retry the test.

11. Click **Next** to continue.
12. On the Select Targets page, select all of your Services Gatekeeper Network Tier servers or clusters.
13. Click **Finish** to save the JDBC data source configuration and deploy the data source to the targets that you selected.
14. To activate your changes, in the **Change Center** of the Administration Console, click **Activate Changes**.

Configure EDR Types for Reports

This section explains how to configure event data records (EDRs) to capture report information using the Administration Console. You can also use the Platform Test Environment or another MBean browser, to make these changes in the **EdrServiceMBean**.

For more information on EDRs and how to configure them. see "Managing and Configuring EDRs, CDRs, and Alarms" in *Services Gatekeeper System Administrator's Guide*

To enable EDR types for reports:

1. Make sure that the Services Gatekeeper Administration Server is running.
2. Start the Administration Console by entering the following URL in your web browser:

   `http://hostname:port/console`

   Where *hostname* is the DNS name or IP address of the Services Gatekeeper Administration Server, and *port* is the address of the port on which the Administration Server is listening for requests (8001 by default).
3. When the login page appears, enter the user name and the password you used to start the Administration Server (you may have specified this user name and password during the Services Gatekeeper installation process), or enter a user name that has been granted one of the default global security roles.
4. If you have not already done so, in the Change Center of the Administration Console, click **Lock & Edit**.
5. Navigate to **OCSG**, then *servername*, then **Container Services**, then **edrService**.
6. Select **setEdrTypes** from the **Operations** menu.
7. Set these EDR types to **true**:
   - **Publish_facade_edr**
   - **Publish_enabler_ecr**
   - **Publish_protocolStack_edr**
8. Click **Invoke**.
9. In the Change Center, select **Release Configuration**.

Deploying the Reports EAR File

To deploy the reports EAR file:

1. Make sure that the Services Gatekeeper Administration Server is running.
2. Start the Administration Console by entering the following URL in your web browser:

   `http://hostname:port/console`

   Where *hostname* is the DNS name or IP address of the Services Gatekeeper Administration Server, and *port* is the address of the port on which the Administration Server is listening for requests (8001 by default).
3. When the login page appears, enter the user name and the password you used to start the Administration Server (you may have specified this user name and password during the Services Gatekeeper installation process), or enter a user name that has been granted one of the default global security roles.
4. If you have not already done so, in the Change Center of the Administration Console, click **Lock & Edit**.
5. In the left pane of the console, select **Deployments**.

6. In the right pane, click **Install**.
7. On the Locate deployment to install and prepare for deployment page, enter the following path in the **Current Location** field and press Enter.

   ```
   Services_Gatekeeper_home/applications
   ```

8. Select one of the following EDR files and click **Next**:
   - For standalone, single server environments, select **edr_to_analytic-single.ear**.
   - For cluster environments, select **edr_to_analytic.ear**.
9. On the Choose targeting style page, select **Install this deployment as an application**, and click **Next**.
10. On the Select deployment targets page, select the Network Tier servers or clusters that comprise your Services Gatekeeper installation, and click **Next**.
11. On the Optional Settings page, accept the defaults, and click **Next**.
12. Click **Next**.
13. Click **Finish**.
14. In the Change Center click **Activate Changes**.
15. Select your Services Gatekeeper domain and choose **Deployments**.
16. In the **Deployments** table, select **edr_to_analytic** and then click **Start** and choose **Servicing all requests**.
17. On the Start Deployments page, click **Yes**.
18. For clustered environments, ensure that the deployed application is started on all of the Network Tier instances in your installation.

Connecting Services Gatekeeper to the Reports Data Source

To connect Services Gatekeeper to the reports data source:

1. Make sure that the Services Gatekeeper Administration Server is running.
2. Start the Administration Console by entering the following URL in your web browser:

   ```
   http://hostname:port/console
   ```

   Where *hostname* is the DNS name or IP address of the Services Gatekeeper Administration Server and *port* is the address of the port on which the Administration Server is listening for requests (8001 by default).
3. When the login page appears, enter the user name and the password you used to start the Administration Server (you may have specified this user name and password during the Services Gatekeeper installation process), or enter a user name that has been granted one of the default global security roles.
4. In the **Domain Structure** tree, expand **OCSG** and select the Network Tier node with **EdrToAnalytic** deployed.
5. On the **Oracle Communications Services Gatekeeper** page, expand **Container Services** and select **EdrToAnalytic**.
6. In the lower panel, select the **Operations** tab and then choose **connectToDatasource** from the **Select An Operation** list box.
7. Click **Invoke**.

8. Ensure that the operation returns a successful connection.

Verifying the Services Gatekeeper Installation

You should now verify your Services Gatekeeper installation. You can do this by using the Services Gatekeeper Platform Test Environment to send messages through Services Gatekeeper and verify that components are communicating and processing traffic.

Where to Go from Here

See "Next Steps".

7

Configuring the Services Gatekeeper Domain

This chapter describes how to configure an Oracle Communications Services Gatekeeper domain.

Before you configure your domain, you must have set the WebLogic Server home path. See "Services Gatekeeper Post-Installation Tasks".

About Configuring Service Gatekeeper Domains

You must configure the domains of all of your servers before you start them. You can use the WebLogic Server Configuration Wizard to manually configure each server in your installation, or you can configure the domain on your Administration Server and then use the **pack** and **unpack** commands provided by Oracle WebLogic Server to package the configuration data for copying to all the other servers. For more information about packing and unpacking configurations, see *Oracle Fusion Middleware Creating Templates and Domains Using the Pack and Unpack Commands*.

After configuring your Services Gatekeeper domains, return to the "Services Gatekeeper Post-Installation Tasks" for instructions on how to start the Services Gatekeeper servers.

The Services Gatekeeper installer copies the **pack** and **unpack** commands to the *Middleware_home*/**wlserver/common/bin** directory.

About the Domain Configuration Tools

You configure your Services Gatekeeper domain with the following tools:

- The WebLogic Server Configuration Wizard, which can be run in GUI mode or console mode.

  If you want to run the Configuration Wizard in GUI mode on Solaris or Linux, the console attached to the machine on which you are configuring the domain must support a Java-based GUI.

- WebLogic Scripting Tool (WLST), which is a command-line tool that provides configuration scripts.

  System administrators and operators use WLST to monitor and manage WebLogic Server instances and domains. The WLST scripting environment is based on the Java scripting interpreter, Jython. For more information about WLST, see *Oracle Fusion Middleware Understanding the WebLogic Scripting Tool*.

Information Requirements

The Configuration Wizard prompts you for the following information about your database:

- The database hostname
- The database instance name
- The database listener port number
- The names of your managed servers
- The database administrative user name and password

Configuring the Domain Using the Configuration Wizard in GUI Mode

The procedure for configuring the domain with the Configuration Wizard follows these steps:

1. Start the Configuration Wizard in GUI mode.
2. Answer the questions in each screen of the Configuration Wizard.

For more information about creating a WebLogic domain by using the Configuration Wizard, see "Creating a WebLogic Domain" in *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard*.

Starting the Configuration Wizard in GUI Mode

To start the Configuration Wizard in GUI mode:

1. Log in to the target system.
2. Go to the *Middleware_home*/**wlserver/common/bin** directory.
3. At a command prompt, enter one of the following:
   - **Windows**:

     ```
     config
     ```

   - **Linux or Solaris**:

     ```
     sh config.sh
     ```

The Configuration Wizard starts and the Configuration Type screen appears. Go to "Configuring the Domain in GUI Mode" and follow the steps for configuring the domain.

Configuring the Domain in GUI Mode

The procedure in this section reflects using the Configuration Wizard in GUI mode, but the screen names are the same in GUI mode and console mode.

Important:

- Configure only one domain at a time.
- Each domain must be created in its own, empty directory.
- If you use IPv6 addresses put the addresses inside square brackets "[]".
- If you will be using CORBA-based functionality that connects to multiple hosts, do not use the value **localhost** in any configurations. Use an actual IP address or fully qualified host name instead.

The Configuration Wizard displays a sequence of screens, in the order listed below. The screens that you will see depend on the type of product configuration template that you select in the Templates screen. To configure your domain, answer the questions in the following screens:

- Configuration Type Screen
- Templates Screen
- Administrator Account Screen
- Domain Mode and JDK Screen
- JDBC Data Sources Screen
- JDBC Data Sources Test Screen
- Advanced Configuration Screen
- Configuration Summary Screen
- Configuration Progress Screen
- Configuration Success Screen

Configuration Type Screen

In the Configuration Type screen:

1. Select **Create a new domain**.
2. In the **Domain Location** field, enter the target domain directory or use the **Browse** button to locate the directory.

   The directory you enter must be empty.
3. Click **Next**.

Templates Screen

In the Template screen:

1. Select **Create Domain Using Product Templates**.
2. In the **Available Templates** area, select only *one* of the following Services Gatekeeper configuration templates.
   - Basic Oracle Communications Services Gatekeeper Domain
   - OCSG Basic HA Configuration
   - OCSG Domain with Access and Network Clusters

- OCSG Domain with Access and Network Clusters with Oracle RAC Configuration
- OCSG Portal Domain

Note: You can configure only one Services Gatekeeper template at a time.

3. Click **Next**.

Administrator Account Screen

In the Administrator Account screen:

1. Enter the main administrator user name.

   This name is used to start the Administration Server and connect to it. The default user name is **weblogic**, which you can use for domain setup and testing. User names are case sensitive. Do not use commas or any characters in the following comma-separated list:

   ```
   \t, < >, #, |, &, ?, ( ), { }
   ```

2. Enter the main administrator password.

   The password is case sensitive and must contain a minimum of eight characters, at least one of which is not alphabetic.

3. Click **Next**.

Domain Mode and JDK Screen

In the Domain Mode and JDK screen:

1. In the **Domain Mode** area, select the appropriate startup mode for your installation:
   - Development Mode
   - Production Mode (This is the only supported mode for 64-bit Solaris environments.)

   If you select **Production Mode**, do not enable SSL unless you have a trusted key. For more information about startup modes, see "Tuning WebLogic Server" in *Fusion Middleware Performance and Tuning for Oracle WebLogic Server*.

2. In the **JDK** area, select the JDK to use for the domain.

   By default, the installer selects the JDK that was used when you installed Services Gatekeeper. Alternatively, you can specify a different JDK.

3. Click **Next**.

JDBC Data Sources Screen

Specify the connection information between Services Gatekeeper and the JDBC data sources (databases).

In the JDBC Data Sources screen:

1. In the table, select the **wlng.datasource** and **wlng.localTX.datasources** checkboxes to configure these data sources simultaneously. To configure these data sources separately, make adjustments in the data source for the transactional data source.

2. Typically, fields you may need to edit include:
   - **Vendor**: The database vendor. The default is **Oracle**. Select **MySQL** if you are using a MySQL database or cluster.
   - **Driver**: The driver for your database type. The available drivers are specific to the vendor value you specified.

     For Oracle databases, the default is Oracle's Driver (Thin) for Instance connections. For non-Oracle RAC domains, use the **non-XA** thin driver for **wlng.localTX.datasource**, and the **XA** driver for **wlng.datasource**.

     For MySQL databases and clusters, select the **com.mysql.jdbc.Driver** for all data sources.
   - **DBMS/Service**: The name of the database you created in "Installing the Database". The default is **SLEE_DB**.
   - **Host Name:** The location of the database, and IP address or **localhost**. The default is **localhost**. To use an IPv6 IP address enter it inside "[]" (square brackets). For example: **[2001:DB8:c8:216:3eff:fe49:c84]**.
   - **Port**: The port number for contacting the database. For Oracle, the default is **1521**. For MySQL, the default is **3306**.
   - **Username**: The Services Gatekeeper user name you created when you installed the database. The default is **SETME_DBUSER**.
   - **Password**: The Services Gatekeeper password you created when you installed the database.
   - **Oracle RAC configuration for data sources**: If you are using Real Application Cluster features, do one of the following:
     - To convert one or more data sources to GridLink Oracle RAC data sources, select **Convert to GridLink**.
     - To convert one or more data sources to Oracle RAC multi-data sources, select **Convert to Oracle RAC multi data source**.
     - To not convert the data sources, select **Don't Convert**.
3. Click **Next**.

JDBC Data Sources Test Screen

The JDBC Data Sources Test screen automatically tests your data source configurations:

- A green check mark displayed in the **Status** column indicates that the configuration is valid.
- A red circle indicates a problem.

  The bottom of the screen shows a short error message to help you troubleshoot the problem. Fix the error and click **Test Selected Connections** to test your data source configurations again.

Click **Next** when you are ready to proceed to the next screen.

Advanced Configuration Screen

The Advanced Configuration screen allows you to perform advanced configuration on the listed items. If you are happy with the current settings, keep all of the checkboxes deselected and click **Next**.

1. In the Advanced Configuration screen, select one or more of the following checkboxes and then click **Next**.
   - Administration Server
   - Node Manager
   - Managed Servers, Clusters and Coherence
   - Deployments and Services

   The next screen that appears depends on the checkboxes that you selected.
2. If you selected **Administration Server**, the Administration Server screen appears.

   Add or change the Administration Server name, listen address, and listen port. Do not enable SSL unless you have a trusted key. Click **Next**.

   Note: If your Services Gatekeeper implementation will communicate using SSL, do not select the **All Local Addresses** listen address menu item and Port **7001**. The Configuration Wizard accepts this option, but attempting to enable SSL later fails. Instead enter a specific local IP address to listen on.

3. If you selected **Node Manager**, the Node Manager screen appears.

   Select the node manager type, enter the node manager credentials, and then click **Next**.
4. If you selected **Manager Servers, Clusters and Coherence**, do the following:
   a. In the Managed Servers screen, add or change the connection information for the managed servers. Each managed server is an instance of Oracle WebLogic Server.

      Click **Add** for each manager server that you want to create. Enter the server name, listen address, and listen port. Do not enable SSL unless you have a trusted key. Click **Next**.
   b. In the Clusters screen, click **Add** for each cluster that you want to create. For example **203.0.113.164:8001, 203.0.113.165:8001**.

      Enter the information about your cluster and frontend. Click **Next**.
   c. In the Coherence Clusters screen, accept or change the default cluster name, but be sure to change the default port number by typing in a new one. Click **Next**.

      Note: Always change default ports for security reasons. In this case it can also help prevent problems when testing multiple Services Gatekeeper implementations.

   d. In the Machines screen, add or change information about each machine.

      In the context of WebLogic Server, a machine is the logical representation of the system that hosts one or more WebLogic Server instances, for the purposes of starting and stopping remote servers using the node manager. In a domain, machine definitions identify a particular, physical piece of hardware and are used to associate a computer with the managed servers it hosts.
5. If you selected **Deployments and Services**, do the following:

a. In the Deployments Targeting screen, target one or more applications to a server or cluster. Select one or more applications in the **Deployments** pane, select one server or cluster in the **Targets** pane, and then click the right arrow button.

b. In the Services Targeting screen, target services to servers or clusters. Select one or more services in the **Services** page, select a server or cluster in the **Targets** pane, and then click the right arrow button.

c. Click **Next**.

Configuration Summary Screen

The Configuration Summary screen displays the previously configured domain settings. Use the **View** drop-down list to choose a category view.

Click **Create** to accept the domain details and start creating the domain.

Configuration Progress Screen

The Configuration Progress screen displays a progress bar that indicates the status of the configuration process. When the configuration progress is complete, click **Next**.

Configuration Success Screen

The Configuration Success screen displays the domain's location and Administration Server URL for accessing the domain.

Click **Finish** to end your configuration session.

Configuring the Domain Using the Configuration Wizard in Console Mode

This section describes how to configure the domain by using the Configuration Wizard in console mode.

Starting the Configuration Wizard in Console Mode

To start the Configuration Wizard in console mode:

1. Log in to the target system.
2. Open a command window.
3. Go to *Middleware_home*/**wlserver/common/bin**.
4. At the prompt, enter one of the following commands and press Enter:
   - Windows:

     ```
     config -mode=console
     ```

   - Linux and Solaris:

     ```
     sh config.sh -mode=console
     ```

   The Configuration Wizard starts in console mode and the Welcome screen appears.

Configuring the Domain in Console Mode

To configure your domain, respond to the prompts in each section by entering the number associated with your choice and pressing **Enter**, or by typing **`Next`** or **`n`** to accept the current selection.

The right arrow (->) indicates the value currently selected. To quit the Configuration Wizard, type **`Exit`** or **`x`** in response to any prompt. To review or change your selection, type **`Previous`** or **`p`** at the prompt.

The screen names and parameters in the Configuration Wizard are the same for both GUI and console modes. See "Configuring the Domain in GUI Mode" for instructions on setting the configuration parameters.

Note: After creating a new domain in console mode, you must copy the *domain_home*/**security/SerializedSystemIni.dat** file from the administration server to the same location on the new domain.

Configuring the Domain Using a WebLogic Scripting Tool Script

This section explains how to configure a Services Gatekeeper domain by using a WebLogic Scripting Tool (WLST) script.

The WLST scripting environment is based on the Java scripting interpreter, Jython. For more information about WLST, see "Using the WebLogic Scripting Tool" in *Oracle Fusion Middleware Understanding the WebLogic Scripting Tool*.

Caution: WLST has a significant learning curve. If you do not know how to use WLST and do not wish to spend the time to become familiar with it, use the Configuration Wizard to set up your domains instead.

Setting Up Your Environment

You must set environment variables for WLST to run properly.

1. Log in to the target system.
2. Open a command window.
3. Go to *Middleware_home*/**wlserver/server/bin**.
4. At the prompt, enter one of the following commands:
   - **Windows**:

     ```
     setWLSEnv.cmd
     ```

   - **Linux and Solaris**:

     ```
     sh setWLSEnv.sh
     ```

Choosing the WLST Domain Setup Script

Services Gatekeeper provides five WLST domain setup scripts and five corresponding domain configuration templates. The scripts are located in *Middleware_home*/**wlserver/common/templates/scripts/wlst**, and the templates are located in *Middleware_home*/**wlserver/common/templates/wls**.

Table 7–1 describes the scripts and their respective domain templates used to configure each type of domain:

Table 7–1 Scripts and Domain Templates

Script	Template	Description
basic-ocsg-ha.py	basic-ocsg-ha-domain.jar	Creates a basic domain with two servers, each with an Access Tier, a Network Tier instance, and a database. Database replication must be set up separately.
ocsg-database-setup.py	ocsg-domain.jar	Creates a basic all-in-one domain typical of development environments.
access-network-cluster.py	ocsg-access-network-domain.jar	Creates a domain with separate access and network clusters.
ocsg-osb-integ.py	ocsg-osb-integ-domain.jar	Creates a domain with separate access and network clusters with the additional data sources that an Oracle RAC installation requires.

Configuring the WLST Script

You must configure the WLST domain setup script to work with your environment. This section describes the configurations you may need to perform.

Configuring Multicluster Settings

Perform this task if you are using a domain setup script other than **wlng-cluster.py**.

If you are setting up the standard version of one of the multi-cluster domains, only a few variables need to be set at the beginning of the script. This procedure describes how to modify the WLST script to set the multicluster settings in the section called **Configuration (INPUT) Parameters**. Example 7–1 shows the necessary configuration parameters that need to be edited for your environment.

Example 7–1 Configuration (INPUT) Parameters Section from Access-Network-rac.py

```
#===========================================================================
# Configuration (INPUT) Parameters
#===========================================================================

# listen address input parameters
# example: hostname can be DNSName or IPAddress

AdminServerListenAddress = "host-admin.bea.com"
AdminServerListenPort    =  7001
NT1ServerListenAddress   = "host-nt1.bea.com"
NT1ListenPort            =  8001
NT2ServerListenAddress   = "host-nt2.bea.com"
NT2ListenPort            =  8001
AT1ServerListenAddress   = "host-at1.bea.com"
AT1ListenPort            =  8001
AT2ServerListenAddress   = "host-at2.bea.com"
AT2ListenPort            =  8001

NTClusterAddress         = "host-nt1.bea.com:8001,host-nt2.bea.com:8001"
ATClusterAddress         = "host-at1.bea.com:8001,host-at2.bea.com:8001"

NTClusterMultiCastAddress = '237.0.0.101'
NTClusterMultiCastPort    = 8050
ATClusterMultiCastAddress = '237.0.0.102'
ATClusterMultiCastPort    = 8050
```

```
# DataSource Settings

# Oracle RAC Node-1 Settings

RACNode1URL      = "SETME_URL"

# Oracle RAC Node-2 Settings

RACNode2URL      = "SETME_URL"

# Database settings

OracleXADriver    = "SETME_XADRIVER"
OracleNonXADriver = "SETME_nonXADRIVER"
DBUser            = "SETME_USER"
DBPassword        = "SETME_PASSWORD"
```

To configure the multicluster settings:

1. Set the listen address and listen port for the Administration Server, the two Access Tier servers, and the two Network Tier servers.
   - Replace the **host*.bea.com** values with either the DNS name or the IP Address of the appropriate servers.
   - Replace the listen port values as necessary. The listen address and port combinations must be unique.
2. Fill in the appropriate listen address and port combinations to assign the servers to the appropriate clusters. The entry should be comma delimited, with no spaces.
3. Fill in the appropriate multicast addresses values for each cluster.
4. If using a configuration script for Oracle RAC deployments:
   - Set the appropriate URLs for each of the Oracle RAC instances.
   - Set the appropriate values for the transactional (XA) and localTX(nonXA) datasources.
5. For non-Oracle RAC deployments:
   - Set the appropriate values for the **wlng.datasource**.
   - Set the appropriate values for the **wlng.localTX.datasource**. The values should be non-XA.
6. (Optional) To use the Administration Console and node manager to start remote servers, change the **NodeManager ListenAddress** values in the **Configure Managed Servers** section by editing the following line for each managed server:

   ```
   set('ListenAddress','localhost')
   ```
7. (Optional) Change the **localhost** value to the correct listen address for your environment.

   The default domain user (weblogic) and password.

Adding Machines and Servers to a Multicluster Configuration

Perform this task if you are using either the **access-network-cluster.py** or the **access-network-rac-cluster.py** domain setup script for cluster configuration and you also want to create additional machines, servers, or both.

Note: You can also add servers and machines using the Administrative Console after you set up your primary Services Gatekeeper domain, which is a simpler way of adding machines and servers.

Using WLST in offline mode, which is the mode that Services Gatekeeper scripts use, allows accessing and updating only those configuration objects that have been previously persisted to a configuration file. All the provided WLST scripts create this configuration file automatically as they run, but each script adds only those objects that are specified in the domain templates they support. If you must add more configuration objects, such as additional managed servers or machines, you must add additional parameters to the script to create them before you can configure them. The specific parameters you add depend on how your installation is set up.

Adding Machines

Use the sample code in Table 7–2 to add machines in the script *before* you assign managed servers to them.

Table 7–2 Code to Add Machines

Comment Section	Code to add	Value

Configure managed servers	`cd('/')`
	`create('new_Machine_5','Machine')`
	`cd('Machine/new_Machine_5')`
	`create('new_Machine_5','NodeManager')`
		Add as many of these statements as you need, replacing *new_Machine_5* with your machine name.

Adding Managed Servers

After you add machines, you can assign managed servers to them. You can also add new managed servers. In the sample code in Table 7–3, a new managed server is created and then assigned to *new_Machine_5*, created in the previous section.

Table 7–3 Code to Create Additional Managed Servers

Comment Section	Statement to edit	Value

Configure managed servers	`cd('/')`
	`create('new_Server_1', 'Server')`
	`cd('Server/new_Server_1')`
	`set('ListenPort', 'port')`
	`set('ListenAddress', address)`
	`set('Machine','new_Machine_5')`
		Create new servers as needed, and set the **ListenAddress**.
		The *new_Server_1* is the name of the new server being created, *port* is the listen port for the server, *address* is the IP address or DNS name of the new server and *new_Machine_5* is the machine to which you are adding the new server.

Setting the NodeManager Listen Address

You must also add a section to configure the **Listen Address** of any new machine (and its node manager) you are adding. The sample code in Table 7–4 shows the WLST statement used to complete this configuration.

Table 7–4 Setting Listen Address for Node Manager

Comment Section	Statement to add	Value

Configure managed servers	`cd('/')`
	`cd('Machine/new_Machine_5')`
	`set('Name','new_Machine_5')`
	`set('Address','address')`
	`cd('NodeManager/new_Machine_5')`
	`set('ListenAddress','new_Server_1')`
	`set('ListenPort','port')`
		One section per added machine is required.
		The *new_Server_1* is the name of the new server being created, *port* is the listen port for the server, *address* is the IP address or DNS name of the new server and *new_Machine_5* is the machine to which you are adding the new server.

Assigning New Managed Servers to a Cluster

You must assign any newly-created managed servers to their appropriate cluster by adding an **assign** command. The sample code in Table 7–5 shows a WLST statement that assigns new managed servers to a cluster.

Table 7–5 Assigning New Managed Servers

Comment Section	Statement to add	Value

Configure a cluster and assign the Managed Servers to that cluster.	`cd('/')`
	`[standard]`
	`assign('Server', 'new_Server_1','Cluster','cluster1')`
		One line per added Managed Server is required.
		The *new_Server_1* is the name of the new server you created and *cluster1* is the cluster you are adding the server to.

Preventing Communication Services from Being Deployed

Perform this task if you know that you will not use one or more communication services and you prefer to prevent them from being deployed.

> **Note:** You can also undeploy communication services at a later time. See *Services Gatekeeper System Administrator's Guide* for information about undeploying communication services.

All communication services consist of two EAR files: an Access Tier file and a Network Tier file. To prevent a communication service from being deployed, add an **unassign** command to your script for both EAR files.

For example, to prevent the PX 3.0 Third Party Call communication service from being deployed, add the following example section to your script:

```
#=====================================================================
```

```
# Unassign applications to target
#=====================================================================
cd('/')
unassign('Application', 'wlng_at_third_party_call_px30#4.0 ', 'Target', 'WLNG_AT_
Cluster')
unassign('Application', 'wlng_nt_third_party_call_px30#4.0 ', 'Target', 'WLNG_NT_
Cluster')
```

Running the WLST Domain Setup Script

After editing the WLST domain setup script, run it using the following command:

```
java weblogic.WLST script_name.py
```

Where *script_name* is the name of the WLST script.

Where to Go From Here

Complete the rest of the Services Gatekeeper post installation tasks, picking up at "Creating JMS Servers for Additional Network Tier Servers".

8

Scaling Services Gatekeeper

This chapter explains how to scale your Oracle Communications Services Gatekeeper implementation up by adding new network tier (NT) and access tier (AT) servers on remote systems.

Understanding Scaling

This section assumes that you have a multi-tier Services Gatekeeper installed and configured and want to add additional processing capability by creating more NT and AT servers on remove systems. You do this by adding additional hardware systems and installing an AT or NT server on each one. You then connect the working Services Gatekeeper administration server to the new servers.

During this process you will install a complete multi-tier implementation of Services Gatekeeper on each system, but you will only start and use the NT or AT server for it.

Adding Remote NT/AT Servers to Services Gatekeeper

This section assumes that you have:

- Installed a Services Gatekeeper implementation on one system, and will use that Services Gatekeeper administration server to manage the new AT and NT servers.
- Created a list of the customizations you made to the system with the administration server. Including any custom configuration files or JAR files.
- Prepared an additional system ready to receive the new AT/NT server by adding the same operating system to it. Each system must have a unique IP address.

To add an additional NT or AT server:

1. Log on to the system to receive the new NT or AT server.
2. Follow the instructions in “"Installing Services Gatekeeper" to install the JDK and a multi-tier implementation of Services Gatekeeper on the system.

   Note: You must install the same version of Services Gatekeeper on every system.

3. Follow the instructions in "Configuring the Services Gatekeeper Domain" to create and configure a domain for the NT or AT server.
4. Add any customized property files or JAR files from the system with the administration server to the new system with the AT or NT server.

5. Log on to the system running your database, and have your database administrator back up the database schema.
6. Log on to the system running your administration server,.
7. Make a backup of the domains.
8. Start the Administration Console.
9. Create a new AT and NT node by navigating to **Environment**, then **Machines** and creating new machines for each NT or AT.
10. Create a new NT or AT server by navigating to **Environment**, then **Servers** and creating a new server.
11. Add the new NT or AT node to the cluster by navigating to **Environment**, then **Clusters** and adding the new AT or NT.
12. Create a new JMS server for the new AT or NT by navigating to **Services**, then **Messaging** ,then **JMS Servers**, and creating a new JMS server.
13. Navigate to **Services**, then **Messaging** ,then **JMS Modules**, and then:
    1. Click **WLNGEDRResource**, then the **Subdeployments** tab.
    2. Click **WLNG_JMS_CLUSTER** and make sure that **JMSServer-NTx** is selected as the target.
14. Navigate to **Environment**, then **Servers**, then *servername*, then **Protocols**, then **Channels** and create a new channel for each AT or NT server.
15. Restart the administration server and all of the AT and NT servers on all systems.
16. Test your services to be sure the administration server communicates with the new AT or NT server.

Automating the Process of Scaling Services Gatekeeper

If you will configure a lot of Services Gatekeeper implementations, or add a lot of NT or AT servers to scale a Services Gatekeeper implementation, you may find it easier to automate the process. See:

- "Installing Services Gatekeeper in Silent Mode" for details on installing Services Gatekeeper installation programmatically.
- “Creating Templates and Domains Using the pack and Unpack Commands” at the WebLogic 12c documentation website for details on creating the domains programmatically:

  `https://docs.oracle.com/cd/E24329_01/web.1211/e24498/toc.htm`

9

Installing Services Gatekeeper Reports

This chapter describes how to set up and install Oracle Communications Services Gatekeeper Reports.

For more information about reports, see “Managing and Configuring Statistics and Transaction Licenses” in *Services Gatekeeper System Administrator’s Guide*.

Overview of Installing Services Gatekeeper Reports

The procedure for installing Services Gatekeeper Reports follows these steps:

1. Install Services Gatekeeper. See "Installing Services Gatekeeper".
2. Perform pre-installation tasks for reports:
   - Install Oracle Business Intelligence.
   - Configure Oracle Business Intelligence for use with Services Gatekeeper.
3. Install Services Gatekeeper Reports.
4. Enable Oracle Business Intelligence Write-back and Iframe Support.
5. Perform post-installation tasks, which include configuring the Services Gatekeeper domain and performing post-installation tasks for Reports. See "Post-Installation Tasks for Reports".

After installing and configuring Services Gatekeeper Reports, Services Gatekeeper reports and statistics can be found in your Oracle Business Intelligence dashboard in the **EDR Analysis Home Page**.

Reports System Requirements

Reports is supported on the following:

- WebLogic Server 10.3.5 or higher
- JDK 1.8 or higher, plus the latest security updates

Installing Services Gatekeeper Reports

The Reports installer runs in GUI mode only. You can create an installation log by using the **-log=***logfilename* parameter on the command line when you run the installer. For more information about creating a log file, see "Creating an Installation Log".

Caution:

- Before continuing, make sure that all components of your Oracle Business Intelligence installation are running, including administration servers, database servers, and any associated domains.
- Services Gatekeeper Reports requires administrator access if you are installing it on a Windows-based host system.

To install Reports, do the following on the system that is hosting Oracle Business Intelligence:

1. If you are installing on a 64-bit system, ensure that a 64-bit JDK or a hybrid 32/64-bit JDK is installed on the target machine.

   If it is not installed, install one. See "Services Gatekeeper System Requirements" for information about supported JDK versions.

2. Run the `java -version` command, or `java -d64 -version` command on platforms using a 32/64-bit hybrid JDK, to ensure that the JAVA_HOME environment variable is set to a 64-bit JDK.

   If JAVA_HOME is not correctly set, set it to point to the correct JDK.

3. Add the **bin** directory of the appropriate JDK to the beginning of the PATH variable definition. For example:

   ```
   PATH=$JAVA_HOME/bin:$PATH
   export PATH
   ```

4. Download the Services Gatekeeper Reports installer from the Oracle software delivery website:

   `https://edelivery.oracle.com/`

5. Change to the directory where you downloaded the installation program.

6. To start the installer, do one of the following:

   - To start the installer on a system that uses a 32/64-bit hybrid JDK, enter:

     ```
     java -d64 -jar ocsg_analytics_generic.jar [-log=logfilename]
     ```

   - To start the installer on a 32-bit system, enter:

     ```
     java -jar ocsg_analytics_generic.jar [-log=logfilename]
     ```

   After the installer starts, the Welcome screen appears.

7. Click **Next**.

   The Installation Location screen appears.

8. In the **Oracle Home** field, enter the full path to your Middleware home directory or use the **Browse** button to locate the directory.

   The Middleware home directory is the central directory for all Oracle products installed on the target system, such as WebLogic Server, Services Gatekeeper, and Services Gatekeeper Reports.

   To see a list of Oracle products that are currently installed in the directory, click **View**.

9. Click **Next**.

   The Installation Type screen appears.

10. Click **Next**.

    The Analytics Parameters screen appears.

11. In the **OBIEE Admin Console** area, enter the following information:

    - **URL**: The URL of the OBIEE Administration Console. For example, **http://server.com:8001/console**.
    - **User Name**: The Oracle Business Intelligence WebLogic domain administrator user name.
    - **Password**: The password for the Oracle Business Intelligence WebLogic domain administrator.

12. In the **Oracle Database for Analytics** area, enter the following information:

    - **Host Name**: The host name of the database server to be used for reports data.
    - **Port**: The port number through which the database host listens.
    - **DBMS/Service**: The name of the database or service hosting the reports data.
    - **User Name**: The user name that will access the reports database.
    - **Password**: The password for the database user.

13. In the **OBIEE stuff for Analytics** area, enter the following information:

    - **RPD file Path**: The local path to the updated Services Gatekeeper RPD file including the name of the RPD file. For example, **/export/home/oracle/edr.rpd**.
    - **RPD file Password**: The password used to access the reports repository.
    - **OBIEE ORACLE_INSTANCE path**: The Oracle instance location, defined when Oracle Business Intelligence was installed. For example, *Middleware_home*/**instances/instance1**.

14. Click **Next**.

    The Prerequisite Checks screen appears.

15. The screen automatically tests your system to ensure that it meets all operating system and JDK software requirements:

    - A green check mark indicates that your system passed the prerequisite check.
    - A red circle indicates a problem. The bottom of the screen shows a short error message to help you troubleshoot the problem. Fix the error and click **Rerun** to perform the prerequisite checks again. To continue the installation without fixing the problem, click **Skip**.

16. Click **Next**.

    The Installation Summary screen appears.

17. Click **Next**.

    The Installation Summary screen appears.

18. Ensure that the listed installation location and feature sets to install are correct.

    If the list is not correct, you can use the **Back** button to make corrections.

    To save the information to a response file so you can install the component later, click **Save Response File** and specify the name and location of the response file.

19. Click **Install** to start the installation.

    The Installation Progress screen appears, and a progress bar indicates the status of the installation process.

20. Click **Next**.

21. When the Installation Complete screen appears, click **Finish**.

    The installer exits.

22. Check the WebLogic Server Administration Console log for errors.

23. Restart your Oracle Business Intelligence instance for the changes to take effect.

24. Complete the procedures in "Enabling Oracle Business Intelligence Write-Back and Iframe Support".

See "Uninstalling Services Gatekeeper Reports" if you need to uninstall Services Gatekeeper reports.

Troubleshooting Services Gatekeeper OBIEE Installation

If you see this error message during the installation process, check the *OBI_home*/**tmp/log_xmf** file for information:

```
oracle.as.install.engine.modules.util.installaction.InstallActionException:
Initial database failed
```

Enabling Oracle Business Intelligence Write-Back and Iframe Support

You must make the following modifications to your Oracle Business Intelligence installation:

- Enable Oracle Business Intelligence write-back support for parameter/value-related reports.
- Enable Oracle Business Intelligence Iframe support to support portal integration.

> **Note:** Make these configuration changes on every Oracle Business Intelligence server in a clustered environment.

To enable Oracle Business Intelligence write-back and Iframe support:

1. Go to *Middleware_home*/**ocsg/ext/analytics**.
2. Copy the write-back template file (**write_back.xml**) to the following location:

   Oracle_instance/**bifoundation/OracleBIPresentationServicesComponent/coreapplication_obips***n*/**analyticsRes/customMessages**

   where *Oracle_instance* is the Oracle Business Intelligence instance path and *n* is replaced by the Oracle Business Intelligence instance number.
3. Open the **instanceconfig.xml** file located at *Oracle_instance*/**config/OracleBIPresentationServicesComponent/coreapplication_obips***n*.
4. Locate the `ServerInstance` element and add it to the `LightWriteback` element:

   ```
   <WebConfig>
      <ServerInstance>
   ```

```
      <LightWriteback>true</LightWriteback>
   </ServerInstance>
<WebConfig>
```

5. Locate the `Security` element and add to it the `InIFrameRenderingMode` element:

```
<Security>
   <InIFrameRenderingMode>allow</InIFrameRenderingMode>
   <!--This Configuration setting...-->
   <ClientSessionExpireMinutes>210</ClientSessionExpireMinutes>
</Security>
```

6. Save your changes and close the file.
7. Restart Oracle Business Intelligence.
8. Access the Oracle Business Intelligence Presentation Services console using a supported browser. The default address is **http://***OBI_host***:9704**, where *OBI_host* is the name of the Oracle Business Intelligence host system.
9. In the Oracle Business Intelligence Presentation Services console, go to **Settings**, then **Administration**, and then **Manage Privileges**.
10. Grant the privilege **Write Back to database** to the appropriate group.

Configure OBIEE Caching For Improved Performance

Services Gatekeeper uses Oracle Business Intelligence Enterprise Edition (OBIEE) for reporting. The OBIEE default behavior is to cache data in the BI server and presentation server caches before making it available to use. You can also use OBIEE data dynamically if your implementation requires it. However, the price is system performance. Dynamic data requires a significantly more bandwidth than cached data.

To turn these OBIEE caches off and use the data dynamically:

- Disable the presentation server caching by adding these entries to the *OBIEE_home***/instances/instance1/config/OracleBIPresentationServicesComponent/coreapplication_obips1/instanceconfig.xml** file:

```
<Cache>
.
    <Query>
.
      <MaxEntries>1</MaxEntries>
.
      <MaxExpireMinutes>-1</MaxExpireMinutes>
.
      <MinExpireMinutes>-1</MinExpireMinutes>
.
      <MinUserExpireMinutes>-1</MinUserExpireMinutes>
.
    </Query>
.
  </Cache>
```

- Disable the BI server cache. You can find example instructions at the OBIEE training web site:

  http://obieetraining11.blogspot.com/2012/08/obiee-11g-disable-caching.html

See the OBIEE documentation for information on these caches.

Uninstalling Services Gatekeeper Reports

This section explains how to remove the Services Gatekeeper extension for OBIEE analytics.

To remove the Services Gatekeeper for OBIEE Analytics:

1. Log on to the system running OBIEE reports.
2. Navigate to the *OBI_home***/oui/bin**.
3. Run the **./deinstall** script.
4. Delete all Services Gatekeeper OBIEE database tables from the staging database. Or you can remove all tables for the staging database user with this command.

   ```
   SQL> delete staging_username User cascade
   ```

   Where *staging_username* is the name of the staging database user using OBIEE reports.
5. Delete the *OBI_home***/ocsg_analytics** directory.

In order to reinstall reports, you need to create a staging database a user to run them.

Where to Go from Here

Perform the remaining post-installation tasks for Reports in "Post-Installation Tasks for Reports".

10

Adding a Communication Service Application

This chapter describes how to add communication service applications to your existing Oracle Communications Services Gatekeeper (OCSG) cloud domain.

Add Communication Service Applications

You can add communication service applications to your existing Services Gatekeeper cloud domain.

> **Note:** The feature applies to a single-tier configuration only. All communication services are installed automatically in a multi-tier configuration.

The tool supports the Linux platform and performs the following operations:

- Gets the necessary application package files from the installer
- Configures the domain to deploy the applications
- Updates the domain to add communication service related services
- Updates the startup script to adjust related settings

The following packages are used by communication services:

Table 10–1 Packages Used By Communication Services

Communication Service	Functionality	Packages Delivered	Optional Feature Set
SMS	Messaging	**wlng_at_sms_parlay_rest.ear** (For one API interface) **wlng_at_sms_px21_soap.ear** (For SOAP interface) **wlng_nt_sms_px21.ear** (MUST be deployed)	Application - SMS (ocsg_app_sms)

Table 10–1 (Cont.) Packages Used By Communication Services

Communication Service	Functionality	Packages Delivered	Optional Feature Set
MMS	Messaging	**wlng_at_multimedia_messaging_parlay_rest.ear** (For one API interface) **wlng_at_multimedia_messaging_px21_soap.ear** (For SOAP interface) **wlng_nt_multimedia_messaging_px21.ear** (MUST be deployed) **wlng_at_multimedia_messaging_mm7.ear** (For SOAP interface, MM7) **wlng_nt_multimedia_messaging_mm7.ear** (MUST be deployed for MM7)	
Payment	Payment	**wlng_at_payment_parlay_rest.ear** (For one API interface) **wlng_at_payment_px30_soap.ear** (For SOAP interface) **wlng_nt_payment_px30.ear** (MUST be deployed)	Application - Payment (ocsg_app_payment)
Terminal Location	Location	**wlng_at_terminallocation_parlay_rest.ear** (For REST interface) **wlng_at_terminal_location_px21_soap.ear** (For SOAP interface) **wlng_nt_terminal_location_px21.ear** (MUST be deployed)	Application - Terminal Location (ocsg_app_termloc)
QoS	Quality of Service	**wlng_at_qos_px40.ear** (For SOAP interface) **wlng_nt_qos.ear** (MUST be deployed)	Application - QoS (ocsg_app_qos)
Application Subscription	Subscription	**wlng_at_app_subscription_rest.ear** (For REST interface) **wlng_nt_app_subscription.ear** (MUST be deployed)	Application - Subscription (ocsg_app_subscription)

Application Installation

The utility first runs a script, **addCommsPack.sh**, that first checks package availability in the current installation and, if necessary, launches a silent installation to a temporary folder to get required files and copy them to the specified OCSG installation location. Only needed files are copied.

Next, a WebLogic Scripting Tool script called **addCommsPack.py** checks the domain configuration to discover missing applications and add them into the deployment. The script works in offline mode only and requires you to shut down OCSG in advance because changing a service requires a shutdown anyway.

Next, **addCommsPack.sh** changes the server setting by manipulating the custom configuration file, **${DOMAIN}/config/custom/nt.xml**. If not presented, the service **oracle.ocsg.protocol.smpp.SMPPServerService** needs to be added.

The script changes the `blockUnknownAPIsflag` in **setDomainEnv.sh** (or **setDomainEnv.cmd**) from true to false.

At this point the domain is ready to use with correctly deployed telecom applications.

Updating Newly Added Packages

The addCommsPack tool updates the installation inventory to add information about the newly added applications as if they are selected in the initial installation, allowing OPatch to update them normally.

After application installation, addCommsPack reads the configuration file `featuresetList` to get information about the feature sets to be added. The tool checks whether a specified feature set is inside the inventory and, if not, updates the following items for each feature set:

- **$ORACLE_HOME/inventory/registry.xml**

  Indicates whether a feature set is selected or not.

- **$ORACLE_HOME/inventory/Components**

  Contains metadata for each component, one folder for one component

- **$ORACLE_HOME/inventory/ContentsXML/comps.xml**

  Contains the information about the installed components

- **$ORACLE_HOME/inventory/refconts/components**

  Contains the reference count for each installed components

- **$ORACLE_HOME/inventory/refconts/feature sets**

  Contains the reference count for each selected feature set

When these items are updated, OPatch can update new applications normally.

The tool updates the following optional feature sets: ocsg_app_mm, ocsg_app_payment, ocsg_app_qos, ocsg_app_sms, ocsg_app_subscription, ocsg_app_termloc.

Additional resource files, including the list of required applications, feature sets, server services, silent installation/deinstallation response files and several template files under the folder **inventory_template** are used for updating the inventory for each optional feature set. All of these files are packed into the installer and copied to **${ORACLE_HOME}/wlserver/common/templates/scripts/addCommsPack/** after installation.

Usage

Follow these steps to use the tool:

1. Prepare a valid single-tier installer file.
2. Set the environment variable `JAVA_HOME` to the JDK location. Otherwise the tool will prompt the user to enter the location.
3. Shut down OCSG.
4. Run the following script and specify the domain location:

   ```
   ${ORACLE_HOME}/wlserver/common/templates/scripts/addCommsPack/addCommsPack.sh
   ${DOMAIN_HOME} ${SINGLETIER_INSTALLER_FILE}
   ```

5. When the script finishes, restart OCSG.

The tool can add any application server service as long as the required information is present in the list files.

You can customize the tool by editing the following files:

- **pkgList**

  Contains the required application package names, for example **wlng_nt_sms_px21.ear**. One line per package. Each line that starts with # is treated as a comment and ignored during processing.

- **serviceList**

  Contains the required service's full class name, for example **oracle.ocsg.protocol.smpp.SMPPServerService**. One line per server service. Each line that starts with # is treated as a comment and ignored during processing.

- **featuresetList**

  Contains the list of optional feature set names to be added to the installation inventory, for example **ocsg_app_sms**. One line per feature set. Each line that starts with # is treated as a comment and ignored during processing.

11

Installing the Platform Test Environment

This chapter describes how to install Oracle Communications Services Gatekeeper Platform Test Environment (PTE).

PTE is a graphical user interface (GUI) tool that you use to test default Services Gatekeeper features and your own custom communication services. For more information about PTE, see "Understanding the Platform Test Environment" in *Services Gatekeeper Platform Test Environment User's Guide*.

Overview of Installing PTE

You download and install the PTE separately from other Services Gatekeeper software.

The procedure for installing Services Gatekeeper PTE follows these steps:

1. Install Services Gatekeeper. See "Installing Services Gatekeeper".
2. Install Services Gatekeeper PTE.
3. Perform post-installation tasks, such as connecting PTE to Services Gatekeeper and configuring your communication services. See "Configuring and Maintaining Your PTE Server Environment" in *Services Gatekeeper Platform Test Environment User's Guide*.

After installing and configuring Services Gatekeeper PTE, you can start it in GUI mode or console mode. For more information, see "Starting the PTE" in *Services Gatekeeper Platform Test Environment User's Guide*.

Installing the PTE in GUI Mode

You can create an installation log by using the **-log=***logfilename* parameter on the command line when you run the installer. For more information about creating a log file, see "Creating an Installation Log".

> **Note:** Services Gatekeeper PTE requires administrator access if you are installing it on a Windows-based host system.

To install PTE:

1. Log in to the target system.
2. Download the Services Gatekeeper PTE installer from the Oracle software delivery website:

   `https://edelivery.oracle.com/`

3. Change to the directory where you downloaded the software.
4. Start the installer:

   ```
   java -jar ocsg_pte_generic.jar [-log=logfilename]
   ```

   After the installer starts, the Welcome screen appears.
5. Click **Next**.

   The Installation Location screen appears.
6. In the **Oracle Home** field, enter the full path to your Middleware home directory or use the **Browse** button to locate the directory.

   The Middleware home directory is the central directory for all Oracle products installed on the target system, such as WebLogic Server, Services Gatekeeper, and Services Gatekeeper PTE.

   To see a list of Oracle products that are currently installed in the directory, click **View**.
7. Click **Next**.

   The Installation Type screen appears.
8. Click **Next**.

   The Prerequisite Checks screen appears.
9. The screen automatically tests your system to ensure that it meets all operating system and JDK software requirements:
   - A green check mark indicates that your system passed the prerequisite check.
   - A red circle indicates a problem. The bottom of the screen shows a short error message to help you troubleshoot the problem. Fix the error and click **Rerun** to perform the prerequisite checks again. To continue the installation without fixing the problem, click **Skip**.
10. Click **Next**.

    The Installation Summary screen appears.
11. Ensure that the listed installation location and feature sets to install are correct.

    If the list is not correct, you can use the **Back** button to make corrections.

    To save the information to a response file so you can install the component later, click **Save Response File** and specify the name and location of the response file.
12. Click **Install** to start the installation.

    The Installation Progress screen appears, and a progress bar indicates the status of the installation process.
13. Click **Next**.
14. When the Installation Complete screen appears, click **Finish**.

Installing the PTE in Silent Mode

This section describes how to install the PTE in silent mode on all platforms.

Use silent mode for installing duplicate installations on multiple machines. You do this by creating and using the **response.rsp** configuration file, and then specifying it as a

parameter during a silent mode installation. When silent mode is used, the installation program does not display any options during the installation process.

To install the PTE in silent mode:

1. Log in to the target system.
2. Download the Services Gatekeeper PTE installer from the Oracle software delivery website:

   https://edelivery.oracle.com/

3. Create a **response.rsp** file in a text editor.
4. Add the following contents to the file:

```
[ENGINE]

#DO NOT CHANGE THIS.
Response File Version=1.0.0.0.0

[GENERIC]

#The oracle home location. This can be an existing Oracle Home or a new Oracle
Home
ORACLE_HOME=Middleware_home

#Set this variable value to the Installation Type selected. For example, .
INSTALL_TYPE=PTE
```

5. Save the file in the directory where you downloaded the PTE package.
6. From the directory where you downloaded the PTE package, enter:

   ```
   java -jar ocsg_pte_generic.jar -silent -responseFile ResponseFile
   ```

   where *ResponseFile* is the full path and name of the Response File. For example, **/home/use/bin/response.rsp**.

The installation proceeds with no prompts.

Where to Go from Here

To finish setting up PTE, perform the post installation tasks described in "Configuring and Maintaining Your PTE Server Environment" in *Services Gatekeeper Platform Test Environment User's Guide*.

Perform the tasks described in "Services Gatekeeper Post-Installation Tasks".

12

Upgrading Services Gatekeeper

This chapter describes how to upgrade Oracle Communications Services Gatekeeper from version 6.1 to version 7.0. Upgrades are supported on the Linux, Solaris, and Windows operating systems.

For more information about supported operating systems, see "Services Gatekeeper System Requirements".

Upgrading from a Pre-6.1 Release

If you are upgrading from an release previous to 6.1, you must first upgrade to Services Gatekeeper 6.1 with patch set 3:

- To upgrade from Services Gatekeeper 5.0.0.1 to 5.1, see the *Services Gatekeeper Installation Guide* for release 5.1.

  http://docs.oracle.com/cd/E36135_01/doc.51/e37539/toc.htm

- To upgrade from Services Gatekeeper 5.1 to 6.0, see the *Services Gatekeeper Multi-tier Installation Guide* for release 6.0.

  http://docs.oracle.com/cd/E50778_01/doc.60/e50756/toc.htm

- To upgrade from Services Gatekeeper 6.0 to 6.1, see the *Services Gatekeeper Multi-tier Installation Guide* for release 6.1.

  https://docs.oracle.com/communications/E64613_01/doc.61/e64621/toc.htm

At this point, you have Services Gatekeeper 6.1 (with patchset 3) installed in your location. To upgrade to Services Gatekeeper 7.0, complete the tasks described in this chapter.

Upgrade Restrictions

The following restrictions apply when upgrading Services Gatekeeper:

- SIP functionality is no longer provided in Gatekeeper 7.0. If you require SIP functionality do *not* upgrade from 6.1.
- If you are using JavaDB as your database, the database upgrade will copy the old database password. Make sure that the domain user password is the same as your 6.1 domain user password, because JavaDB will use that old password as its own.
- Do not make configuration changes (such as changing Services Gatekeeper MBean attributes) during the upgrade process until all of the servers in the cluster have been upgraded. This is especially important for new configuration options.

Services Gatekeeper servers ignore settings that are not understood, and the local configuration file may not be updated properly.

- During the upgrade, the location of the old Services Gatekeeper installation and the location of the new Services Gatekeeper installation must be in different directories.
- Set the number for processes/sessions at database level in Oracle to suit your requirements. If your Oracle database is dedicated to Services Gatekeeper only, then 500 processes are recommended. If your Oracle database is shared between Services Gatekeeper and other systems, set the number for processes to between 1000 and 2000.
- For Portal analytics, your customers must re-submit the password for the Analytics server.

Placeholders Used in This Chapter

In addition to the placeholders described in "Placeholders Used in this Guide", Table 12–1 describes the directories and values that are specific to upgrading.

Table 12–1 Upgrade-Specific Placeholders

Placeholder	Description
new_Middleware_home	The new Middleware home directory is the directory under which you install Services Gatekeeper 7.0. Middleware home is the repository for common files that are used by Oracle Communications service delivery products such as Services Gatekeeper, WebLogic Server, and Java Development Kit.
new_Services_ Gatekeeper_home	The directory in which the new version of the Services Gatekeeper software is installed. By default, this is *new_Middleware_home***/ocsg**.
old_version	The two-digit version number, without periods, of the existing Services Gatekeeper version to be upgraded. For example, **70** represents version 7.0.
new_version	The two-digit version number, without periods, of the new Services Gatekeeper version to which you are upgrading. For example, 70 represents version 7.0.
new_domain_home	The directory in which you create the new Services Gatekeeper domain. By default, this is a subdirectory of the *new_Middleware_ home***/user_projects/domains** directory.

Handling Case Sensitivity in MySQL

If you are using Oracle database as data source of Services Gatekeeper, please ignore this section.

If you are using MySQL with Oracle Linux as the database for Services Gatekeeper, then complete the steps in this section.

Set the SQL Table name identifier, **lower_case_table_names**, to be case insensitive when starting **mysqld**.

To do so:

1. Go to the location where **my.cnf** file is located. By default, my.cnf is typically located in **/etc**.
2. Open the **my.cnf** file at a terminal to edit it:

```
sudo vi $mysql/my.cnf
```

3. Locate this start of the [mysqld] section seen as:

   ```
   [mysqld]
   ```

4. Directly below this line, add the following entry:

   ```
   lower_case_table_names = 1
   ```

5. Save the file.
6. Restart mysql:

   ```
   sudo /etc/init.d/mysql restart
   ```

7. To verify the change, run this command:

   ```
   mysqladmin -u root -p variables
   ```

Upgrade and Rollback for 6.1 to 7.0

Oracle Communications Services Gatekeeper (OCSG), version 7.0, provides procedures to upgrade from version 6.1 to 7.0 and to roll back from 7.0 to version 6.1, if necessary.

> **Note:** SIP-related functionality is not supported in OCSG 7.0. If you need this functionality, you should not migrate from version 6.1.

These are the summary steps to upgrade to OCSG 7.0:

1. Install a complete OCSG 7.0, create a new 7.0 domain, and point to a new database.
2. Recreate all of the domain configurations, including both OCSG and cluster-specific configurations.
3. Stop the old 6.1 cluster.
4. Migrate your data from your 6.1 database to a new 7.0 database and adjust the schema and the data, particularly your passwords.
5. Start a new 7.0 cluster.

Database Migration

The principal task in migrating from OCSG 6.1 to 7.0 is migrating from the old database to the new database. The database migration consists of these four steps, which you must complete manually:

1. Copy data from the old database to the new database.
2. Adjust the table schema in the new database to match the 7.0 definitions.
3. Adjust the data in the tables to match 7.0 usage.
4. Re-encrypt the password records in the tables to match the new domain key.

Different database types provide different solutions for copying or cloning the database and additional third-party tools can also do the job effectively. This section suggests some possible procedures.

Note: When copying the database, the existing 6.1 cluster must be stopped.

Copying a Java Database

The Java database provides both online and offline tools for backup.

The `SYSCS_UTIL.SYSCS_BACKUP_DATABASE()` procedure locks the database and performs the copy. It takes a string argument that specifies the location in which to back up the database. Typically, this is the full path to the backup directory. Relative paths are interpreted as relative to the current directory rather than relative to the **derby.system.home** directory.

For example, you would use the following statement to specify a backup location of **c:/mybackups/2004-06-01** for the currently open database:

```
CALL SYSCTL_UTIL.SYSCS_BACKUP_DATABASE('c:/mybackups/2004-06-01')
```

Note: Use forward slashes as the path separator in SQL commands.

The `SYSCS_UTIL.SYSCS_BACKUP_DATABASE()` procedure puts the database in a state in which you can safely copy it and then copies the entire original database directory, including data files, online transaction log files, and jar files to the backup directory. Files that are not within the original database directory, such as **derby.properties**, are not copied.

To make an offline copy of a Java database, you can simply copy the database directory to the destination directory. If the database size is large, Oracle recommends for efficiency to first create a zip file of the database directory and then unzip it in the destination directory.

Note: Before making an offline copy, you must first shut down the Java database. Then copy the database directory directly.

Copying a MySQL Database

MySQL provides the commands `mysqldump` and `mysqlimport` to copy a database. You can also use the `mysql` command. Here are the steps:

1. On the MySQL command line, create a new database for the OCSG 7.0 domain.
2. Use the `mysqldump` command to dump the data from the old OCSG 6.1 database.
3. Use the `mysql` command to import data to the newly created database.

The following example illustrates:

```
mysqldump --host 10.182.12.146 --port 3306 -user ocsgtest --password upgrade61 >
dump.sql
mysql --host 10.182.12.146 --port 3306 --user ocsgtest --password upgrade70 <
dump.sql
```

Copying an Oracle Database

The Oracle database provides a set of tools that enable a user to clone database data. The following example uses the `exp` and `imp` commands to copy the database:

1. In SQL *Plus, create a new database user and password.

2. Grant `EXP_FULL_DATABASE` privilege to the OCSG 6.1 database user.
3. Grant `CONNECT, RESOURCE`, and `IMP_FULL_DATABASE` to the new user.
4. Run the `exp` command to export all of the data and schema information from the OCSG 6.1 database to a file.
5. Run the `imp` command to import all of the data and schema information from the exported file.

The following example illustrates:

```
grant EXP_FULL_DATABASE to upgrade61;
grant IMP_FULL_DATABASE to upgrade70;
exp upgrade61/123456 FILE=clone.dmp owner=upgrade61
imp upgrade70/123456 FILE=clone.dmp fromuser=upgrade61 touser=upgrade70
```

Upgrade Procedures

The procedure to upgrade from release 6.1 to release 7.0 makes the following assumptions:

- OCSG 6.1 is installed in the directory **`$OCSG_61_INSTALL`**
- The OCSG 6.1 domain is installed in the directory **`$DOMAIN_61_INSTALL`**
- OCSG 7.0 is installed in the directory **`$OCSG_70_INSTALL`**
- The OCSG 7.0 domain is installed in the directory **`$DOMAIN_70_INSTALL`**
- If you use the Java database, the OCSG 7.0 cluster's administration server is located in the same host where the 6.1 administration server was located.

The Java Database Procedure

If you are using a Java database, use the following migration procedure:

1. On the OCSG 6.1 cluster, if you use API management, approve or reject all pending requests on the partner portal.
2. Install the latest Java Development Kit (JDK).
3. Install OCSG 7.0 in the directory **`$OCSG_70_INSTALL`**.

> **Note:** Because copying the database also copies the old database password, the new OCSG domain user password must be the same as for the 6.1 installation because the Java database also reuses this password.

4. Configure the OCSG 7.0 cluster. If necessary, use the customized offline configuration.

> **Note:** Do not start the new OCSG 7.0 servers at this point.

5. On the host where the administration server is located, in the directory $OCSG_70_INSTALL/ocsg/upgrade/, unzip the file **migration.zip**, which extracts the directory **6.1**. In the directory $OCSG_70_INSTALL/ocsg/upgrade/6.1, change the mode of all shell files to make them executable.
6. Stop the OCSG 6.1 cluster and the Java database for the cluster.

7. On the host where the administration server is located, in the directory $OCSG_61_INSTALL/wlserver/common/derby, execute the following command:

   ```
   cp -R gatekeeper $OCSG_70_INSTALL/wlserver/common/derby
   ```

   Note: If the database is large, you can zip and unzip the database the database to improve the efficiency of the copy.

8. On the host where the administration server is located, in the directory **$DOMAIN_70_INSTALL/bin**, execute the following command to start the Java database for the 7.0 cluster:

   ```
   ./dbController.sh start
   ```

9. On the host for the administration server, in the directory **$OCSG_70_INSTALL/ocsg/upgrade/6.1**, run the `runConfigurationMigration.sh` script. It will prompt you for the following information:
   - OCSG 7.0 cluster's database type (oracle, mysql, or javadb). Input: "javadb"
   - OCSG 7.0 cluster's database host
   - OCSG 7.0 cluster's database port (default is 1527)
   - OCSG 7.0 cluster's database name. Input "gatekeeper"
   - OCSG 7.0 cluster's database user. Input "gatekeeper"
   - OCSG 7.0 cluster's database password. Input the domain administrator's password.
   - OCSG 6.1 domain home directory
   - OCSG 7.0 domain home director
10. When the script completes, you will see the message, "Congratulations! OCSG 6.1 migration completed successfully."
11. Start the OCSG 7.0 cluster.
12. After startup, perform customized online configuration, if any.
13. Extend the domain, if needed.

Migration is complete.

Oracle Database Procedure

If you are using the Oracle database, use the following migration procedure:

1. On the running OCSG 6.1 cluster, if you use API management, approve or reject all of the pending requests on the partner portal.
2. Install the latest Java Development Kit (JDK).
3. Install OCSG 7.0 in the directory $OCSG_70_INSTALL and create a new 7.0 database for the installer.
4. Configure the OCSG 7.0 cluster and do any offline configuration, if needed.

   Note: Do not start the new OCSG 7.0 servers in this step.

5. On the host where the administration server is located, in the directory **$OCSG_70_INSTALL/ocsg/upgrade/**, unzip the file **migration.zip**, which extracts the directory **6.1**. In the directory **$OCSG_70_INSTALL/ocsg/upgrade/6.1**, change the mode of all shell files to make them executable.
6. Stop the OCSG 6.1 cluster and the Java database for the cluster.
7. On the host where the OCSG 6.1 database is located, copy the database. Refer to "Copying an Oracle Database" for detailed information.
8. On the host for the administration server, in the directory **$OCSG_70_INSTALL/ocsg/upgrade/6.1**, run the `runConfigurationMigration.sh` script to adjust the data, inputting the following information:
   - OCSG 7.0 cluster's database type (oracle, mysql, or javadb). Input: "oracle"
   - OCSG 7.0 cluster's database host
   - OCSG 7.0 cluster's database port
   - OCSG 7.0 cluster's database name
   - OCSG 7.0 cluster's database user
   - OCSG 7.0 cluster's database password. Input the domain administrator's password
   - OCSG 6.1 domain home directory
   - OCSG 7.0 domain home directory
9. Start the OCSG 7.0 cluster.
10. After startup, perform customized online configuration, if any.
11. Extend the domain, if needed.

Migration is complete.

The MySQL Database Procedure

If you are using a MySQL database, use the following migration procedure:

1. On the running OCSG 6.1 cluster, if you use API management, approve or reject all of the pending requests on the partner portal.
2. Install the latest Java Development Kit (JDK).
3. Install OCSG 7.0 in the directory $OCSG_70_INSTALL and build up the 7.0 database.
4. Configure the OCSG 7.0 cluster and do any offline configuration, if needed.

> **Note:** Do not start the new OCSG 7.0 servers in this step.

5. On the host where the administration server is located, in the directory $OCSG_70_INSTALL/ocsg/upgrade/, unzip the file **migration.zip**, which extracts the directory **6.1**. In the directory $OCSG_70_INSTALL/ocsg/upgrade/6.1, change the mode of all shell files to make them executable.
6. Stop the OCSG 6.1 cluster and the Java database for the cluster.
7. On the host where the OCSG 6.1 database is located, copy the database. Refer to "Copying a MySQL Database" for detailed information.

8. On the host for the administration server, in the directory **$OCSG_70_INSTALL/ocsg/upgrade/6.1**, run the `runConfigurationMigration.sh` script to adjust the data, inputting the following information:
   - OCSG 7.0 cluster's database type (oracle, mysql, or javadb). Input: "mysql"
   - OCSG 7.0 cluster's database host
   - OCSG 7.0 cluster's database port
   - OCSG 7.0 cluster's database name
   - OCSG 7.0 cluster's database user
   - OCSG 7.0 cluster's database password. Input the domain administrator's password
   - OCSG 6.1 domain home directory
   - OCSG 7.0 domain home directory
9. Start the OCSG 7.0 cluster.
10. After startup, perform customized online configuration, if any.
11. Extend the domain, if needed.

The migration is complete.

Rolling Back Version 7.0

To roll back an upgrade to OCSG 7.0, simply stop the 7.0 OCSG cluster together with the associated database and start the old OCSG 6.1 cluster.

Upgrading Services Gatekeeper Reports

To upgrade Services Gatekeeper Reports, complete the tasks described in this section.

Reset Text-based API Passwords

If you use the Services Gatekeeper API management features, and your APIs use Text-based security, you need to reset their passwords after upgrading the Services Gatekeeper software. This is required by a change to the Services Gatekeeper security requirements, so you can reset the same usernames and passwords if you want to. Start Services Gatekeeper, and open the Partner and API Management GUI.

For each API that uses **Text** security, set a username and password.

Reports Upgrade Process

To upgrade Services Gatekeeper Reports:

1. Go to the *new_Middleware_home***/ocsg/applications** directory.
2. Deploy the **edr_to_analytic.ear** file targeted to the Network Tier cluster.
3. Deploy the **analytics_proxy.war** file targeted to the Application Tier cluster.

For Portal analytics, your customers must re-submit the password for the analytics server.

13

Uninstalling Services Gatekeeper

This chapter describes how to uninstall Oracle Communications Services Gatekeeper and its components.

Uninstalling Services Gatekeeper Components in GUI Mode

To uninstall Services Gatekeeper components in GUI mode:

1. Go to the *Middleware_home***/oui/bin** directory and enter the following in a command window:

   ```
   ./deinstall.sh
   ```

2. If you have multiple Oracle products installed in your *Middleware_home* directory, the Distribution to Uninstall screen appears.

   Perform the following to specify the Oracle product to uninstall:

   a. From the **Select Distribution to Uninstall** list, select the software you want to uninstall. For example, select one of the following:

      Oracle Communications Services Gatekeeper 6.0.0.0.0

      OCSG Platform Test Environment 6.0.0.0.0

      OCSG Application Test Environment 6.0.0.0.0

      OCSG Analytics 6.0.0.0.0

   b. Click **Uninstall**.

3. From the Welcome screen, click **Next**.

   The Deinstallation Summary screen appears.

4. Verify that the list of feature sets to deinstall is correct.

   To save the information to a response file so you can uninstall the components later, click **Save Response File** and specify the name and location of the response file.

5. Click **Deinstall**.

   The Deinstallation Progress screen appears, and a progress bar indicates the status of the uninstall process.

6. When the uninstallation process is complete, click **Next**.

   The Deinstallation Complete screen appears.

7. Click **Finish**.

The uninstaller exits.

8. (If you are removing Reports information) Perform these steps to remove all reports and reports user information:

   a. Log on to the Analytics database and run this command:

      ```
      delete user User cascade
      ```

   b. Remove the *Gatekeeper_home*/**ocsg_analytics** directory.

Uninstalling Services Gatekeeper Components in Silent Mode

This section describes how to uninstall Services Gatekeeper components in silent mode on all platforms.

Use silent mode to uninstall duplicate installations on multiple machines. You do this by creating and using a **response_uninistall.rsp** configuration file, and then specifying it as a parameter during a silent mode uninstallation. When silent mode is used, the program does not display any options during the uninstall process.

To uninstall Services Gatekeeper components in silent mode:

1. Create a text file on your target system.
2. Add the following contents to your file:

   ```
   [ENGINE]
   #DO NOT CHANGE THIS.
   Response File Version=1.0.0.0.0

   [GENERIC]

   #This will be blank when there is nothing to be de-installed in distribution
   level
   SELECTED_DISTRIBUTION=Oracle Communications Services Gatekeeper~6.0.0.0.0

   #The oracle home location. This can be an existing Oracle Home or a new Oracle
   Home
   ORACLE_HOME=Middleware_home
   ```

   where SELECTED_DISTRIBUTION is set to the component to uninstall. This parameter is required only if multiple applications are installed in *Middleware_home*. The following list shows the values for Services Gatekeeper components:

   - Oracle Communications Services Gatekeeper~6.0.0.0.0
   - OCSG Application Test Environment~6.0.0.0.0
   - OCSG Platform Test Environment~6.0.0.0.0
   - OCSG Analytics~6.0.0.0.0

3. Save the file with the name **response_uninstall.rsp**.
4. Go to the *Middleware_home*/**oui/bin** directory and enter the following in a command window:

   ./deinstall.sh -silent -responseFile *ResponseFile*

   where *ResponseFile* is the full path and name of the **response_uninstall.rsp** file. For example, *Middleware_home*/**oui/bin/response_uninstall.rsp**.

   The uninstall process completes with no prompts.

5. (If you are removing Reports information) Perform these steps to remove all reports and reports user information:
   a. Log on to the Analytics database and run this command:

      ```
      delete user User cascade
      ```

   b. Remove the *Gatekeeper_home*/**ocsg_analytics** directory.

If the uninstall procedure completes successfully, you will see a response similar to the following:

```
Launcher log file is /tmp/OraInstall2014-11-06_01-46-10PM/launcher2014-11-06_
01-46-10PM.log.
Starting Oracle Universal Installer

Checking if CPU speed is above 300 MHz.   Actual 2893.030 MHz    Passed
Checking swap space: must be greater than 512 MB.   Actual 15826924 MB    Passed
Checking if this platform requires a 64-bit JVM.   Actual 64    Passed (64-bit not
required)
Checking temp space: must be greater than 300 MB.   Actual 136360 MB    Passed


Preparing to launch the Oracle Universal Installer from /tmp/OraInstall2014-11-06_
01-46-10PM
Java HotSpot(TM) 64-Bit Server VM warning: ignoring option MaxPermSize=512m;
support was removed in 8.0
Log: /tmp/OraInstall2014-11-06_01-46-10PM/deinstall2014-11-06_01-46-10PM.log
Setting ORACLE_HOME to /home/oracle/
Copyright (c) 1996, 2015, Oracle and/or its affiliates. All rights reserved.
Starting silent deinstallation...
-----------20%----------40%----------60%----------80%--------100%

The uninstall of Oracle Communications Services Gatekeeper 6.0.0.0.0 completed
successfully.
Logs successfully copied to /export/oraInventory/logs.
```

14

Next Steps

This chapter provides information about initial system administration tasks that you must perform after you have completed all Oracle Communications Services Gatekeeper installation and post-installation tasks.

Configuring Services Gatekeeper

You can now proceed to configuring Services Gatekeeper itself. The configuration tasks are different depending on the Services Gatekeeper features you will use. See "Administration Overview" in *Services Gatekeeper System Administrator's Guide* for more information.