Oracle® Communications ASAP

Installation Guide

Release 7.4.1
G13665-02
March 2025

Oracle Communications ASAP Installation Guide, Release 7.4.1

G13665-02

Contents

8 Installing the Order Control Application Client

9 ASAP Post-Installation Tasks

10 Verifying the ASAP Installation

11 Upgrading ASAP

12 Uninstalling ASAP

Preface

This guide explains how to install and upgrade Oracle Communications ASAP and OCA Clients.

Audience

This document is intended for system administrators, system integrators, and other individuals who need to install, upgrade, and work with ASAP.

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.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

ASAP Installation Overview

This chapter provides an overview of the installation process for Oracle Communications ASAP.

Overview of the ASAP Installation Procedure

Before installing ASAP, the following applications must be installed:

- Platform
- Oracle Database and client
- Oracle WebLogic Server

The following is an outline of the installation procedure for ASAP:

1. Plan your installation. Planning your installation involves:
   - Determining the scale of your implementation; for example, is it a small test system, or a large production system. You should assess the scale of the network that ASAP must manage.
   - Assessing the physical machine you need for the ASAP components that you have selected.
   - Planning the system topology; for example, standalone ASAP deployment.
2. Review system requirements. System requirements include:
   - System software requirements, such as operating system (OS) versions and OS patch requirements.
   - Hardware requirements, such as disk space.
   - Information requirements, such as IP addresses and host names.
3. Configure the operating system as described in "Configuring the Operating System ."
4. Create and configure the Oracle Database instance as described in "About Creating and Configuring the Oracle Database."
5. Download and install the Oracle Client as described in "Installing and Configuring Oracle Database and Client."
6. Install and configure Oracle WebLogic Server as described in "About Configuring Oracle WebLogic Server."
7. Use a RedHat Package Manager (RPM) for Installing ASAP.
8. Use the configuration file to set up all the required parameters for Configuring ASAP.

   You must provide the following connection information during installation:
   - WebLogic Server details
   - Database, or Oracle RAC database server details
   - ASAP schema user details
9. Configure ASAP using the configureASAP script.

10. Perform post-installation configuration tasks, such as deploying ASAP cartridges.
11. Verify the installation.

ASAP Configuration Options

You can configure ASAP in the following ways:

- **Console mode:** Use the console mode to manually provide passwords when prompted for on the console.
- **Silent mode:** Use the silent mode for automatically configuring ASAP using the configuration file.

Ensuring a Successful ASAP Installation

You must be familiar with the following before you begin the installation:

- UNIX operating system
- Oracle WebLogic Server administration
- Oracle Database administration

Additionally, you should have experience installing Java-related packages.

Oracle recommends that the installation and configuration of the Oracle database be performed by an experienced database administrator.

Follow these guidelines:

- As you install each component, for example, the Oracle database and WebLogic Server, verify that the component installed successfully before continuing the installation process.
- Pay close attention to the system requirements. Before you begin installing the application, ensure your system has the required base software. In addition, ensure that you know all of the required configuration values, such as host names and port numbers.
- Make a note of any new configuration values as you create them. You will be required to enter configuration values later in the procedure.

Downloading the ASAP Software

To download the software from the Oracle software delivery website (`https://edelivery.oracle.com`):

1. Select the ASAP software for your platform from the Oracle software delivery website.
2. Download the software.
3. Extract the components that you need from the downloaded file.
4. Follow the installation procedure for each component that you want to install.

Directory Placeholders Used in This Guide

Table 1-1 lists and describes placeholders used in this guide.

Table 1-1 Directory Placeholders Used in This Guide

Placeholder	Directory Description
ASAP_home	The directory into which the ASAP software is installed.
ASAP_src	The directory into which the ASAP source installation files are extracted.
Middleware_home	The location where the Oracle Middleware product is installed. This directory contains the base directory for WebLogic Server and the **oracle_common** directory, among other files and directories.
WebLogic_home	The base directory for the WebLogic Server core files. It is located in the *Middleware_home* directory; for example, *Middleware_home***/wlserver**.
domain_home	The directory that contains the configuration for the domain into which ASAP is installed. The default location is *Middleware_home***/user_projects/domains/***domain_name* (where *domain_name* is the name of the ASAP domain), but it is frequently set to another directory at installation.
Oracle_home	The location which contains the Oracle database server software.
Oracle_client_home	The location which contains the Oracle client software.

2

Planning Your ASAP Installation

This chapter describes the hardware, operating system, software, server, and database requirements for installing Oracle Communications ASAP.

About Planning Your ASAP Installation

ASAP is comprised of a series of applications, each with its own database schema which are installed on an Oracle WebLogic Server domain. ASAP connects with an Oracle database to store all relevant information.

About Test Systems and Production Systems

Create test systems to support the following activities:

- Familiarize yourself with ASAP functionality.
- Investigate the ASAP Server implementation size for your production system.
- Determine the ASAP server deployment options for your production system.
- Determine the number and size of the Oracle Database tablespaces that your production system will require.
- Determine the memory requirements for the individual ASAP server schemas.
- Test WebLogic Server functionality and deployment options for test and production environments.
- Develop new network activation (NA) cartridges or service activation (SA) network cartridges, or integrate and customize pre-existing ASAP cartridges.
- Investigate and implement possible ASAP customization requirements for the ASAP SRP or NEP.

Create production systems after fully testing and integrating ASAP functionality according to your network requirements.

Types of Implementations

This section provides details on ASAP server and client implementations.

ASAP Server Implementation

This section provides details on ASAP server implementation size classifications, memory, and ASAP server disk space requirements.

ASAP Server Implementation Size

The ASAP implementation provided in this section are for approximate sizing purposes. Your implementation requirements may vary. The ASAP pre-tuned configuration is 10 to 20+ orders per second, 500,000 to 1M+ orders per day.

The complexity of any custom code extensions can also affect ASAP performance. For complete details on pre-tuned ASAP system configuration, see the appendix on tuning ASAP in the *ASAP System Administrator's Guide*.

ASAP Server Memory

Table 2-1 lists example memory requirements for ASAP servers on UNIX running in an Oracle virtual machine. These example memory requirements also apply to other operating systems supported by ASAP (see "Hardware Requirements" for supported operating systems).

Table 2-1 ASAP Server Memory Requirements

Application	Memory Requirements	Description
SARM Server	65 MB	The amount of memory required for the SARM depends on: • Number of concurrent work orders (WOs) being processed • Complexity of ASAP service modeling based on the number of Common Service Description Layer (CSDL) commands per work order and the number of Atomic Service Description Layer (ASDL) commands per CSDL • Number of target network elements • Number of service request processors (SRPs) for event notification • Internal resource for SARM configuration (for example, the number of threads)
Java SRP Server	256 MB	The Java SRP is managed through WebLogic Server and requires a minimum of 256 MB in production.
NEP Server – asc_nep	40 MB on each server.	The NEP memory usage depends on: • Number of NEs managed by the NEP • Number of communication devices used by the NEP
NEP server – Java process	128 MB	Minimum production size

The Oracle database server also has memory requirements. For sizing guidelines, consult the Oracle Database documentation.

ASAP Disk Space

The disk space requirements for the test and the production system differ. These space requirements described in the following sections are for the file systems required by the ASAP installer only and do not account for additional software (for example, the Oracle Database or WebLogic Server).

You must ensure that you have enough disk space, appropriately allocated, before beginning the installation process.

Test System Disk Space

Test systems have simple disk space requirements because they do not have to accommodate a production work order stream or a full array of network element mappings. Production systems have more complex disk space requirements than test systems.

Test systems are used to build and test support for network elements and build automated interfaces into ASAP for upstream systems. Ensure that you have enough disk space available on your machines for all of the requirements as listed in Table 2-2.

Table 2-2 ASAP Component Requirements

Components	Disk Space
ASAP Installer TAR file size by platform	Oracle Linux: 212 MB
ASAP core components size by platform	Oracle Linux: 598 MB
ASAP logs	100 MB

Production System Disk Space

Production systems are used against live network elements to process real work orders.

The sizing of production systems depends on variables such as daily transaction volume and amount of historical data to be maintained. If your production environment generates many log files due to high volumes of work orders, allocate additional space. Consult with your UNIX administrator for assistance.

Disk I/O speed can be a factor in performance, especially if many log entries are being written. Oracle recommends that you distribute your database and index segments over multiple disks as described in "ASAP Server Hardware Requirements" for small, medium, and large system for increased performance.

For detailed information on tuning ASAP, see *ASAP System Administrator's Guide*.

Determining the Number of Network Elements per Network Element Processor

The number of Network Elements (NEs) per Network Element Processor (NEP) required is dependent on:

- Whether the NE interface is asynchronous or synchronous.
- Whether the NE can support independent concurrent connections.
- The speed of the NE interface (for example, a slow MML response time or fast EMS interface).
- The number of requests to that NE per day. This is usually expressed in terms of work orders per day although a more accurate measure is the number of ASDL commands per day.
- The complexity of custom code extensions or Java code.

For more information about adding NEP servers, see the *ASAP Server Configuration Guide*. For more information about mapping NEs to NEP servers, see the *ASAP Cartridge Development Guide*.

Table 2-3 shows NEs classified according to their anticipated loads and provides a rough estimate of the number of NEs typically allocated to a single NEP.

Table 2-3 Suggested NE to NEP Ratios

NE Classification	Number of Orders per Day per NE	Average Number of NEs per NEP
Idle	Fewer than 10	30 to 100
Normal	Between 10 and 100	15 to 30

Table 2-3 (Cont.) Suggested NE to NEP Ratios

NE Classification	Number of Orders per Day per NE	Average Number of NEs per NEP
Busy	More than 100	5 to 15

The number of NEs that can be managed by a particular NEP is limited only by available system resources. The NEP allocates threads to manage each NE and device used to interface to the NEs. For example, an NEP managing 20 NEs where each NE has two dedicated connections would require 20 + (2 x 20) = 60 threads within the NEP.

Table 2-4 shows the typical memory requirements for NEPs managing different numbers of NEs.

Table 2-4 Memory Requirements per NEP

NEs per NEP	Approximate Memory Requirements
0	7 MB
10	10 MB
20	15 MB
50	20 to 35 MB
500	50 to 70 MB

> **Note:**
>
> The example of 500 NEs per NEP is provided for reference purposes only. Your configuration may differ.

Table 2-5 shows an example for a client with 200 NEs.

Table 2-5 NE/NEP Configuration Example

NE Classification	Number of NEs	NEs per NEP	Number of NEPs Required	Memory (MB)	Cumulative Number of NEPs	Cumulative Memory (MB)
Idle	200	50	4	4 x 50 = 200	4	200
Normal	200	20	10	10 x 15 = 150	14	350
Busy	100	10	10	10 x 10 = 100	24	450
Totals	400	N/A	N/A	N/A	24	450

System Deployment Planning

This section provides details about:

- ASAP Server Process Deployment Options
- Oracle Database Deployment Options Supported by ASAP
- Oracle WebLogic Server Deployment Options Supported by ASAP

- ASAP Reliability Deployment Planning

ASAP Server Process Deployment Options

Install the ASAP environment on a UNIX machine. This UNIX machine runs the ASAP server process (with the exception of the JSRP that runs in the WebLogic server).

All ASAP server processes will reside on one UNIX machine (see "Deploying ASAP to One UNIX Machine")

Deploying ASAP to One UNIX Machine

When you install ASAP on one machine, ASAP must be able to access an Oracle database instance and WebLogic Server instance. Oracle database instance can be co-resident with the ASAP environment, or located on different machines but WebLogic Server instance must be co-resident with the ASAP environment.

A typical configuration consists of ASAP co-resident with the WebLogic Server instance connected to a remote Oracle database instance.

Oracle Database Deployment Options Supported by ASAP

Each ASAP server process has a database schema. The ASAP server schemas reside in asingle instance or in an Oracle Real Application Clusters (RAC) database instance and are associated with one or more tablespace.

ASAP can be:

- Deployed with a single Oracle database instance (see "Deploying ASAP to a Single Database")
- Deployed with an Oracle RAC database instance (see "Deploying ASAP to a Real Application Clusters Database")

Deploying ASAP to a Single Database

You can deploy ASAP to a single Database instance. This database can be co-resident with ASAP or located on a remote machine.

Deploying ASAP to a Real Application Clusters Database

You can enhance Oracle Communications ASAP reliability using the ASAP server configurations and Oracle RAC database. With these added configurations, the ASAP server does not shut down if it loses connection with the database. The ASAP behavior is same for a connection failure during initial connection and during normal operation.

In ASAP installations with a single database instance and an Oracle RAC system, ASAP servers can be configured to wait for a specific period of time before the connection to the database is lost. You can also configure the number of attempts to establish the database connection and the interval between attempts. When the ASAP server reconnects, it uses transparent application failover (TAF) to reconnect to a preconfigured secondary instance or to the same instance of a single instance database. It creates a new connection identical to the connection established on the original instance. The connection properties are the same as the original connection.

Oracle WebLogic Server Deployment Options Supported by ASAP

Several ASAP functions deploy to a WebLogic Server instance. ASAP can be:

- Deployed with an administrator WebLogic Server instance (see "Deploying ASAP to a WebLogic Administration Server")
- Deployed with an administrator WebLogic Server instance and a managed Weblogic Server instance (see "Deploying ASAP to a WebLogic Managed Server")

Deploying ASAP to a WebLogic Administration Server

For development environments, Oracle recommends that you deploy ASAP to a single WebLogic administration server.

Deploying ASAP to a WebLogic Managed Server

For production environments, Oracle recommends that you deploy ASAP to a WebLogic Server instance with an administration server and a managed server.

ASAP Reliability Deployment Planning

There are many configuration solutions to address availability and the right configuration option is often a balance between recovery time requirements and solution costs. System unavailability and downtime can occur for various reasons with most people associating downtime with hardware failure. While hardware failure is a contributing factor to system unavailability, other factors include network outages, software error and human error.

ASAP was designed for maximum system availability and application resiliency to various failure conditions. The ASAP control daemon (Control Server) process monitors other ASAP processes and restarts them if they fail.

To protect against network or disk failures, ASAP may be deployed using the following subsystem components:

- Mirrored dual-ported data disks to protect the application from loss of critical configuration data.
- Backup or redundant network interfaces to assure network connectivity to application clients.
- Backup or redundant networks to assure network connectivity to application clients.
- Backup or redundant Power Distribution Units to guard against system power outages.

ASAP is also certified for Oracle RAC to protect against database server failures. The ASAP Control Server monitors the connection to the Oracle Database and maintains database connectivity in the event of an Oracle RAC database node failover.

In addition to these considerations, ASAP provides the following availability recommendations:

- Configuring a Cold Standby ASAP Server
- Configuring ASAP Clusters

Configuring a Cold Standby ASAP Server

To offer additional reliability in the event of a hardware failure on the server running the ASAP SARM, ASAP may be deployed in a cold standby environment. A cold standby environment refers to a type of availability solution that allows ASAP to run on one server at a time (active/passive). The active server processes the orders while the standby server is installed with the same configuration as the active server.

Implementations of cold standby ASAP servers have been accomplished by systems integrators with solutions that are tailored to the customers' needs.

> **Note:**
>
> ASAP supports an active/passive cold standby deployment configuration. ASAP does not support active/passive warm or hot standby deployment configurations, or an active/active deployment configuration.

Configuring ASAP Clusters

For high availability functionality customers may run ASAP in a clustered environment or third-party software. High availability using clustering is non-native to ASAP. ASAP does not support deployment into a clustered WebLogic Server environment using WebLogic Clustering support.

Implementations of ASAP using third-party high availability clustering software have been accomplished by systems integrators with solutions that are tailored to the customers' needs.

Database and Client Planning

The ASAP installer uses tablespaces to create data and index segments for ASAP server schemas.

For test environments, the ASAP installer can create data and index segments for all server schemas within a single Oracle Database tablespace. In production environments, more complex configurations may be required to enhance performance. For example, you may need a separate tablespace for the SARM data and index segments, or separate tablespaces for all the ASAP servers data and index segments. You must create the required tablespaces prior to the installation of the ASAP Server software.

The ASAP server data stored in the Oracle Database tablespaces implements the following schemas:

- Service Activation Request Manager Server Schema

  The SARM tablespace tables contain a large amount of dynamic information about past, present, and future ASAP requests.

- Control Server Schema

  Two types of data are generated dynamically: performance data and event logs. The amount of performance data generated is configurable by modifying the time interval of sampling. If the time interval of generating data is two hours or more, and the performance data and event logs are purged every two days, 40 MB is sufficient.

  For information about modifying the time interval of sampling, see *ASAP System Administrator's Guide*.

- Admin Server Schema

  This schema contains work order performance information, such as how long it takes to complete a CSDL and ASDL, how many CSDLs are processed, and the ASDL queue size for each NE.

- Network Element Processor Server Schema

  This optional schema stores static data, the amount of which will not grow.

For more information on Oracle Database versions supported for this release, see *ASAP Compatibility Matrix*.

ASAP Oracle Database Tablespace Sizing Requirements

Use the information in this section to plan how you will create and configure the Oracle Database tablespaces required for the ASAP server software.

Recommended Tablespace and Redo Log Sizes for ASAP

Table 2-6 lists the recommended tablespace and redo log sizes for test environments and large production environments.

Table 2-6 Recommended Tablespace and Redo Log Sizes for ASAP

Environment	Recommended Size
Test	5 GB for the entire database. Each individual ASAP test environment requires at least 75 MB. The actual size of these tablespaces and redo log files depends on the number of testing environments expected on each Oracle Database instance.
Production	The actual disk space usage is dependent on the size of logs and completed order retention as well as order size. Create three 1 to 5 GB redo log files on three separate drives.

> **Note:**
>
> Your Oracle DBA must create these tablespaces before installing ASAP. While it is possible to create individual tablespaces for each environment, you can combine the tablespace requirements for many test environments into fewer and larger tablespaces.

Suggested Data to Index Segment Size Ratio

Although data and index segments grow at different rates, Oracle recommends that there should be one-to-one size ratio between your data and index segments.

More Detailed Database Size Estimates

Table 2-7 should be used only as a guideline to determine your tablespace sizing.

Table 2-7 Sample ASAP Tablespace Sizing Requirements

Database	Space (KB/WO)	Description
SARM	8 to 20 KB per WO	The SARM tablespace tables contain a large amount of dynamic information about past, present, and future ASAP requests. The data size per work order can remain uniform from client to client, however sizes have become increasingly varied, specifically with next generation services. The maintenance of extensive NE history information increases the average work order size in the tablespace. Also, the data gathered from the NEs (for example, on queries) can be substantial and change from client to client, depending on business rules, OSS infrastructure, and so on. The example configuration provided is for a medium range ratio of KB per work order. For a medium-sized telco with 50,000 work orders per day, the SARM tablespace requirements can be calculated as follows: • Total number of work orders per day: 50,000 • Total reserved space: 700,000 KB • Total space per work order: 14 KB • Data and log space per work order: 10 KB • Index space per work order: 4 KB For more details about estimating tablespace sizing requirements, see "Sample Service Activation Request Manager Tablespace Requirements."
CTRL	Not applicable per work order	The CTRL (Control Server) tablespace tables maintain dynamic information related to: • Application process performance • System events and alarms This dynamic information can be subject to different archiving and purging policies. Therefore, if the system configuration generates a large number of system events and alarms and has a long purge interval, more space can be required. The CTRL tablespace tables do not contain any dynamic information related to the work order volume.
ADM	Not applicable per work order	The ADM (Admin) server tablespace tables maintain the following statistical information on ASAP processing: • Work order statistics • CSDL statistics • ASDL statistics • NE statistics • ASDL/NE statistics The size of these tables depends on the following factors: • Poll period between the retrieval of statistical information by the Admin Server from the SARM • Number of CSDLs, ASDLs, and NEs in the system • Archive and purge interval of the data in these tables. These intervals can be quite large as the data can be used for reporting purposes.
NEP	0 to 5 KB per WO	As the NEP tablespace tables maintain only static information, not dynamic, they are usually quite small. If dynamic information is generated and maintained in the NEP tablespace tables, then this space requirement can increase. The NEP data requirements are approximately 10 to 20 MB.

Sample Service Activation Request Manager Tablespace Requirements

This section contains sample SARM database sizes.

The average work order size is an important factor in determining the tablespace size requirements. Oracle recommends that you have a rough estimate of the sizing requirements on the ASAP application databases.

Table 2-8 contains the results of a test suite of 80 work orders processed by ASAP to produce rough SARM sizing requirements.

Table 2-8 SARM Work Order Sizing Estimates

Name	Row Total	Reserved	Data Segment Size	Index Segment Size	Unused
tbl_asap_stats	0	80 KB	2 KB	8 KB	70 KB
tbl_asdl_log	311	62 KB	42 KB	2 KB	18 KB
tbl_info_parm	88	32 KB	10 KB	10 KB	12 KB
tbl_srq_csdl	191	32 KB	10 KB	6 KB	16 KB
tbl_srq_log	3482	526 KB	496 KB	8 KB	22 KB
tbl_srq_parm	4782	368 KB	142 KB	208 KB	18 KB
tbl_wrk_ord	80	48 KB	14 KB	4 KB	30 KB

> **Note:**
>
> Estimates for **tbl_info_parm** assume that limited information is brought back from the NE and stored in the SARM. If queries are performed, the tablespace needs to be sized accordingly.

The work order details that produced the results for of the dynamic SARM tables listed in Table 2-8 are specified below:

- 80 work orders
- 2.4 CSDLs per work order
- 3.9 ASDLs per work order
- 60 parameters per work order
- 44 work order log entries per work order

For such orders, the size breakdown is:

- Total data size: 726 KB
- Total index size: Consider the index segment size as approximately equal to the data index size. Index size may vary depending on logging levels, operations, and so on
- Size of switch history log data within the log data is approximately 400 KB, index size is 6 KB.

This implies the space required for each work order, including the switch history data is 12 KB/ work order.

Without any switch history, these values are reduced to approximately 8 KB.

> **Note:**
>
> These figures indicate that for simple residential orders, an average of 12 KB per work order is required in the SARM database. This estimate assumes fairly conservative switch history for each work order. Larger volumes of switch history have an impact on this estimate.

Estimating Service Activation Request Manager Tablespace Size Requirements

Most of the space in the tablespace is required for data generated by the SARM during provisioning. The following estimations are used for tablespace sizing:

- The average work order (or a transaction) takes 15KB.
- Add or deduct 5KB for a bigger or smaller work orders.

As an example, consider a system that processes 10,000 immediate work orders/day, and has an average of 2,000 future-dated work orders. Completed work orders are purged every three days, and failed work orders are handled promptly.

The SARM schema sizing can be calculated follows:

- 10,000 x 15KB=150MB
- 2,000 x 15KB=30MB
- for 3 days: 150 x 3 + 30 = 480MB
- add about 15% on the top for peak time and static data: 480 x 1.15 = 552MB
- take 550MB

> **Note:**
>
> Some orders, such as those that query NEs, can generate a large amount of data that requires additional space per order to store in the SARM.

Java Service Request Processor Tablespace Requirement

Most of the JSRP data is incorporated into the SARM tablespace. See "Sample Service Activation Request Manager Tablespace Requirements."

Oracle Client

The Oracle Client is required for ASAP components to communicate with the Oracle Database. This client must be installed and configured before you install ASAP.

For more information about the Oracle Client versions supported for this release of ASAP, see *ASAP Compatibility Matrix*.

WebLogic Server Planning

This section provides details on WebLogic Server domain configurations supported by ASAP.

WebLogic Server Domain Options

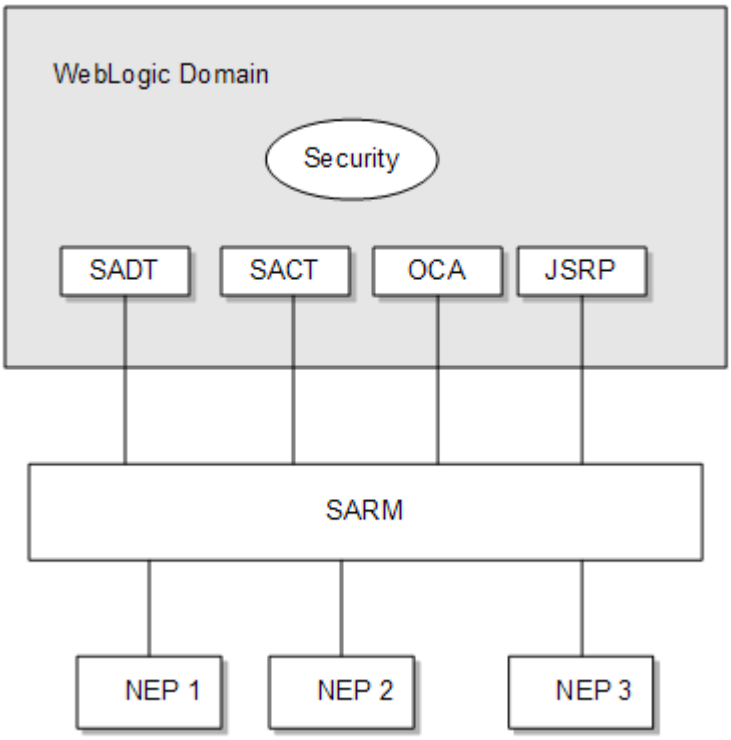
ASAP supports the following domain configurations:

- For test environments: one administration server
- For production environments: one administration server and one managed server

An ASAP domain consists of one administration and one optional managed WebLogic Server instance and their associated resources possibly distributed over two machines. A managed server obtains its configuration from the administration server upon startup. Consequently, the administration server should be started before the managed server.

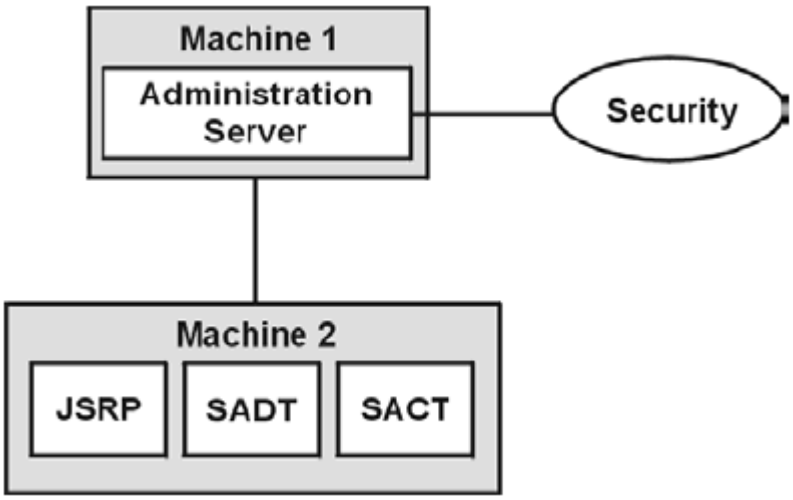
Figure 2-1 shows a WebLogic domain within ASAP.

Figure 2-1 WebLogic Domain Within ASAP



In a test environment, you can create a domain that consists of an administration server and deploy ASAP components to this server instance. However, in a production environment, Oracle recommends that the administration server reside on a dedicated physical machine and ASAP components be deployed to the managed server only (see Figure 2-2). The managed server is assigned to a physical machine. The administration server connects to a machine's node manager, which the administration server uses to monitor, start, stop, and restart a domain's managed server.

Figure 2-2 Administration and Managed Server



The node manager monitors the health of all servers on a machine and controls the restarting of failed servers.

Each domain contains a configuration file – *domain_home*/**config/config.xml** – that stores changes to managed objects so that they are available when WebLogic Server is restarted.

Do not manually modify the **config.xml** file. If you must make configuration changes use the WebLogic Server Administration Console.

For more information about **config.xml**, see the latest Oracle Fusion Middleware WebLogic Server documentation.

Configuring Domain Networks

In a simple server setup, you assign a single network address and port number to each WebLogic Server instance.

You can configure the domain with multiple port numbers to improve performance and solve common networking problems. These port numbers allow you to:

- Separate administration traffic from application traffic in a domain by creating an administration channel.
- Improve network throughput by using multiple NICs with a single WebLogic Server instance.
- Designate specific NICs or multiple port numbers on a single NIC for use with specific WebLogic Server instances.
- Physically separate external, client-based traffic from internal, server-based traffic in a domain.
- Prioritize network connections that servers use to connect to other servers in a domain.

If your domain contains a managed server that is running on a different machine or if your domain contains clients that use different protocols, you can use network channels. Using a single custom channel with multiple servers simplifies network configuration for a domain – changing a channel configuration automatically changes the connection attributes of all servers that use the channel.

You can use multiple channels to segment network traffic by protocol, listen ports, or any other channel configuration property. For example, you can use two channels with a single server to tailor the default connection properties for secure vs. non-secure traffic. You can also use multiple channels to separate external, client traffic from internal, server-to-server traffic.

Most WebLogic Server installations use one or more of the following common types of channels:

- **Default Channel** – WebLogic Server automatically creates a default channel to describe the listen address and listen port settings associated with the ServerMBean. You can view the default channel configuration during server startup.
- **Administration Channel** – You can define an optional administration port to separate administration traffic from application traffic in your domain. When you enable the administration port, WebLogic Server automatically generates an Administration Channel based on the port settings.
- **Custom Channels** – A custom channel is a channel that you define and apply in a domain (rather than a channel that WebLogic Server automatically generates).

A network channel defines the basic attributes of a network connection to WebLogic Server including:

You configure network channels as distinct entities in the Administration Console, and then assign one or more channels to servers in a domain. The server instances to which you assign a channel use the port numbers and protocol configuration associated with the channel, instead of the default network configuration.

For information about using network channels, see the latest Oracle Fusion Middleware WebLogic Server documentation.

Node Manager

Node Manager is a standalone Java program provided with WebLogic Server that you can use to:

- Start remote Managed Server.
- Restart Managed Server that have shut down unexpectedly (for example, due to a system crash, hardware reboot, or server failure).
- Automatically monitor the health of Managed Server and restart server instances that have reached the "failed" health state.
- Shut down or force the shut down of a Managed Server that has failed to respond to a shutdown request.

For more information about using the node manager, see the latest Oracle Fusion Middleware WebLogic Server documentation.

3
ASAP System Requirements

This chapter describes the hardware, operating system, software, server, and database requirements for installing Oracle Communications ASAP.

Hardware Requirements

The number and configuration of the machines that you employ for your ASAP installation depends on the scale and the kind of deployment you have planned according to your networks.

ASAP Server Hardware Requirements

The ASAP hardware requirements described in this section are based on the number of ASDLs processed per second.

Table 3-1 provides recommended system hardware requirements for ASAP installed on an Oracle Linux platform.

Table 3-1 Hardware Requirements for Linux

Software Components	Recommended System Linux 2 threads per core
ASAP Server and WebLogic Server	4 cores, 16 GB RAM
Oracle Database	4 cores, 32 GB RAM
External Storage for Oracle Datafiles	28 x 73 GB (RAID 1+0)

Hardware Sizing Requirements

Table 3-2 describes the allocation of hardware sizing.

Table 3-2 Hardware Sizing Requirements

Number of ASDLs	Recommended Size
Less than 40 ASDLs/second	1 OCPU and 16 GB RAM
Up to 80 ASDLs/second	2 OCPU and 32 GB RAM

High Availability Sizing Requirements

Each ASAP instance in a high availability environment must be sized to handle the full load of orders when one ASAP system is unavailable.

Table 3-3 High Availability Sizing Recommendations

ASAP Components	Recommended Size
Order Balancer virtual machine	1 OCPU and 16 GB RAM
ASAP	1 OCPU and 16GB RAM

For full availability, you can have 2 Order Balancer systems with a HTTP load balancer in front of the Order Balancer for full high availability.

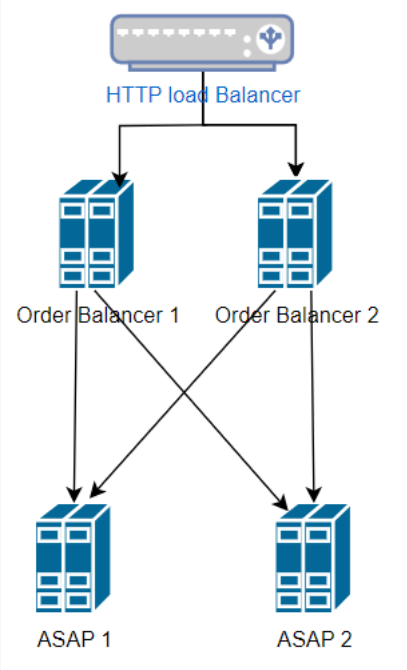
With this sizing recommendation, each ASAP instance can process 20 ASDLs/seconds in normal peak load, however it can handle the fulll load of 40 ASDLs/second when the other system is unavailable.

The sizing recommendation applies to both traditional and cloud native installations.

For more information on how to set up ASAP for high availability, see "Setting Up ASAP for High Availability" in ASAP System Administrator's Guide.

Figure 3-1 describes the high availability environment with the HTTP Load Balancer in front of the Order Balancer.

Figure 3-1 High Availability with HTTP Load Balancer



Software Requirements

For details about the software required to support the ASAP components for traditional deployment. See "ASAP Compatibility Matrix".

Information Requirements

During ASAP installation, you are required to enter configuration values, such as host names and port numbers. This section describes the information that you will be required to provide during the installation process. You define some of these configuration values when you install and configure the Oracle database, client, and WebLogic Server.

Oracle Database Information

Oracle database information is required for the configuration properties file. See "Configure ASAP Database" and "Configure ASAP Database Table space" in "Sample Configuration File" for the list of database parameters required for the ASAP installation.

Table 3-4 lists schema details that you are required to provide during installation.

Table 3-4 Schema user Information

Configuration Parameter	Description
Server Port Numbers	ASAP configures ports based on the values set in the **sampleASAPConfiguration.properties** file. You can provide your own values during the configuration process. You can select ports for the following servers (where *envID* represents your ASAP environment ID): • **SARM***envID* • **CTRL***envID* • **NEP**_*envID* • **OCA**_*envID* • **JSRP***envID* **- Sending WO** • **JSRP***envID* - **Receiving WO event** • **NEP***envID* • **NEP***envID* - *jListener*

For more information about creating tablespaces, see "About Creating and Configuring the Oracle Database."

WebLogic Server Connection Information

Table 3-5 lists WebLogic Server connection details that you are required to provide during installation. The values for the information types are read from the sample configuration (**sampleASAPConfiguration.properties**) file. See, Sample Configuration File for more details on the sample configuration file and its parameters.

> **Note:**
>
> You must enter the values for the information types in the **sampleASAPConfiguration.properties** file.

Table 3-5 WebLogic Server Connection Information

Information Type	Description
Host Name of Oracle WebLogic Server	You provide the host name for the WebLogic Server instance to define it uniquely for the specific purpose of installing, and working with, ASAP (for example, **localhost**).
Port Number of Oracle WebLogic Server	This is the port number assigned to this specific the WebLogic Server (for example **7001**).
User Name of Oracle WebLogic Server Administrator	Your WebLogic Server user name. You define this name when you create an Oracle WebLogic Server domain.
Password of Oracle WebLogic Server Administrator	You define this password along with the user name when you create an Oracle WebLogic Server domain.

4

Configuring the Operating System

This chapter describes how to configure operating systems supported by Oracle Communications ASAP.

About Configuring the Operating System

ASAP can be installed on Oracle Linux. This section describes procedure relating to configuring operating system and other required applications for use with the ASAP server software.

Before proceeding, the appropriate version of the UNIX operating system must be installed on any machines that will be running ASAP server software, and the machines must be configured to be part of your network. See *ASAP Compatibility Matrix* for a list of supported operating systems.

> **Note:**
>
> In order to install and configure the UNIX environment for ASAP as described in this section, you must be familiar with the UNIX operating system. You should know how to configure users and groups, understand shells, and be able to manipulate file and directory permissions. You also need to understand a number of other UNIX system administration concepts.

Configuring a UNIX ASAP Group and User

To support installation and running of ASAP, each ASAP environment that you create must have at least one dedicated UNIX ASAP group and one dedicated ASAP user account within that group. You may choose to have additional ASAP users in the ASAP group, depending on your specific implementation.

Prior to installing ASAP, you must create a UNIX ASAP group and an ASAP user. The ASAP group must be the primary group of the ASAP user.

Creating the UNIX ASAP Group

This group is required to coordinate access to files that are needed to develop and test ASAP. All UNIX users of ASAP must belong to this group.

> **Note:**
>
> Oracle recommends that you use UNIX groups to coordinate the read/write access required for ASAP. You can use any other name for the group that is appropriate for the project, as long as the ASAP and environment users belong to that group.

Create a group for ASAP with the following members:

- ASAP user account
- root

Creating the UNIX ASAP User Account

Each ASAP environment must have a dedicated UNIX user account. For development purposes, you may have multiple ASAP environments. In this case, you must define multiple users. In a production environment, you generally have only one environment and therefore only one group and user.

Oracle recommends for configuration if you set the home directory of the ASAP user to be the same directory where you plan to install the ASAP software. Take this into account when you set up the home directory.

Create the ASAP user and password, and make the user's primary group the newly created ASAP group. Additional ASAP user accounts may be needed depending on your implementation.

> **Note:**
>
> You must use this ASAP user account to create the directory where ASAP will be installed. The installer uses the ASAP user account to access the UNIX system and must have write privileges on the destination install directory.

Checking /tmp and /var/tmp Directory Status

You must ensure that the directories **/tmp** and **/var/tmp** have their permissions set so that they are writable by all ASAP users. To do this:

1. Check the permissions on the **/tmp** directory:

   ```
   cd /
   ls -l | grep tmp
   ```

   Check that the permissions string has the appropriate settings for the ASAP group. Use chmod to update the permissions if needed. For example:

   ```
   chmod a+rwx tmp
   ```

2. Check the permissions on the **/var/tmp** directory.

   ```
   cd /var
   ls -l | grep tmp
   ```

Configuring the Kernel

There are certain UNIX kernel configuration parameters that require tuning so that both the database and ASAP function correctly. The most important aspect of UNIX kernel configuration is to provide an environment in which the database server can handle hundreds of concurrent connections.

The following are the principal areas that require tuning:

- Shared memory

  Oracle Database requires the ability to share large amounts of memory. Typically, the potential shared memory size should be 100 MB or greater.

- Open files per process

  The RDBMS server is expected to manage a large number of concurrent connections. These connections constitute open file descriptors to the RDBMS Server. The maximum number of file descriptors dictates the maximum number of concurrent connections that can be managed by the RDBMS Server. To support multiple ASAP environments, the UNIX kernel should be configured to support an appropriately large number of open files per process.

  > **Note:**
  >
  > Consult with your operating system administrator to determine a value.

  In addition, some operating systems use the concept of soft and hard file limits. Ensure that both limits are set appropriately.

To update the UNIX kernel configuration:

1. Using the information below and in "Configuring the Kernel" update the UNIX kernel parameters.
2. Reboot the machine after you have made the changes.

For more information, see the *ASAP System Administrator's Guide*.

> **Note:**
>
> When ASAP is deployed on the same platform as the Oracle Database or WebLogic Server instance, use the highest kernel value recommended in either ASAP, Oracle Database, or WebLogic Server documentation.

Setting Linux Kernel Parameters

Oracle recommends that the kernel parameters shown in Table 4-1 are set to values greater than or equal to the value shown.

The procedure following the table describes how to verify and set the values manually.

> **Note:**
>
> The kernel parameter and shell limit values shown in the following section are recommended values, and should be modified after installing Oracle 19*c* Client for Linux and before installing ASAP.

Table 4-1 lists and describes the recommended kernel parameter settings for supported Linux systems.

Table 4-1 Kernel Parameter Configuration

Parameter	Recommended Value	File and Description
semmsl semmns semopm semmni	250 3200 32 128	**/proc/sys/kernel/sem** This file contains the four numbers that define the limits of the System V IPC semaphores. These fields are: • semmsl: The maximum number of semaphores per semaphore set. • semmns: The system-wide limit on the number of semaphores in all semaphore sets. • semopm: The maximum number of operations that may be specified in a semop(2) call. • semmni: The system-wide limit on the maximum number of semaphore identifiers.
shmall	268435456	**/proc/sys/kernel/shmall** This file contains the system-wide limits on the total number of pages of System V IPC shared memory.
shmmax	4294967295	**/proc/sys/kernel/shmmax** This file can be used to query and set the run-time limit on the maximum System V IPC shared memory segment size that can be created. Oracle recommends 4 GB for optimum system performance.
msgmax	65536	**/proc/sys/kernel/msgmax** The msgmax tunable specifies the maximum allowable size of any single message in a System V IPC message queue, in bytes. msgmax must be no larger than msgmnb (the size of the queue).
msgmnb	65536	**/proc/sys/kernel/msgmnb** The msgmnb tunable specifies the maximum allowable total combined size of all messages queued in a single System V IPC message queue at one time, in bytes. The default is 16384.
msgmni	2878	**/proc/sys/kernel/msgmni** The msgmni tunable specifies the maximum number of system-wide System IPC message queue identifiers (one per queue). The default is 16.
threads_max	131072	**/proc/sys/kernel/threads-max** Gets and sets the limit on the maximum number of running threads system-wide.
file-max	327679	**/proc/sys/fs/file-max** This file defines a system-wide limit on the number of open files for all processes.

Table 4-1 (Cont.) Kernel Parameter Configuration

Parameter	Recommended Value	File and Description
queues_max	256	**/proc/sys/fs/mqueue** The queues_max is a read/write file for setting and getting the maximum number of message queues allowed on the system.
msgsize_max	8192	**/proc/sys/fs/mqueue** The msgsize_max is a read/write file for setting and getting the maximum message size value. This message queue attribute is used when creating any message.
msg_max	32767	**/proc/sys/fs/mqueue** The msg_max is a read/write file for setting and getting the maximum number of messages in queue value.

Table 4-2 provides the commands to display the current values of the kernel parameters. Make a note of these values and identify any values that you must change.

Table 4-2 Commands to Display Kernel Parameters

Parameter	Command
semms1, semmns, semopm, and semmni	# /sbin/sysctl -a | grep sem Displays the value of the semaphore parameters in the order listed.
shmall, shmmax, and shmmni	# /sbin/sysctl -a | grep shm Displays the details of the shared memory segment sizes.
msgmnb, msgmni, and msgmax	# /sbin/sysctl -a | grep msgm Displays the details of the System V IPC message queue parameters.
threads-max	# /sbin/sysctl -a | grep threads-max Displays the maximum number of threads possible system-wide.
file-max	# /sbin/sysctl -a | grep file-max Displays the maximum number of open files throughout the system.
queues_max	# /sbin/sysctl -a | grep queues_max Displays the maximum number of message queues allowed on the system.
msgsize_max	# /sbin/sysctl -a | grep msgsize_max Displays the maximum message size value.
msg_max	# /sbin/sysctl -a | grep msg_max Displays the maximum number of messages in a queue value.

If the value of any kernel parameter is different from the recommended value, perform the following:

1. Using any text editor, create or edit the **/etc/sysctl.conf** file, and add or edit lines similar to the following:

> **Note:**
>
> Include lines for kernel parameters values that you want to change. For the semaphore (`kernel.sem`) parameter, you must specify all values. However, if any of the current values are larger than the recommended values, then specify the larger value.

```
kernel.sem = 250 32000 32 128
kernel.shmall = 268435456
kernel.shmmax = 4294967295
kernel.msgmax = 65536
kernel.msgmnb = 65536
kernel.msgmni = 2878
kernel.threads-max = 131072
fs.file-max = 327679
fs.mqueue.queues_max = 256
fs.mqueue.msgsize_max = 256
fs.mqueue.msg_max = 32767
```

By specifying the values in the **/etc/sysctl.conf** file, these values persist when you restart the system.

2. Enter the following command to change the current values of the kernel parameters:

   ```
   # /sbin/sysctl -p
   ```

   Review the output from the command to verify that the values are correct. If the values are incorrect, edit the /etc/sysctl.conf file, then enter this command again.

3. Enter the command `/sbin/sysctl -a` to confirm the values are set correctly.

4. After updating the values of the kernel parameters in the **/etc/sysctl.conf** file, either restart the computer, or run the command `sysctl -p` to make the changes in the **/etc/sysctl.conf** file available in the active kernel memory.

Required Linux Packages for ASAP on Linux 8

You must install the following Linux 64-bit packages before you install database client, WebLogic server, and ASAP. The ASAP installer fails to start if there are any missing packages.

The required Linux packages are:

- bc
- binutils
- elfutils-libelf
- fontconfig
- glibc
- glibc-devel
- csh
- ksh
- libaio

- libasan
- liblsan
- libX11
- libXau
- libXi
- libXrender
- libXtst
- libgcc
- libibverbs
- libnsl
- librdmacm
- libstdc++
- libxcb
- libvirt-libs
- make
- policycoreutils
- policycoreutils-python-utils
- smartmontools
- sysstat

5

Installing and Configuring Oracle Database and Client

This chapter describes the tasks relating to the Oracle database that you perform before installing Oracle Communications ASAP.

About Creating and Configuring the Oracle Database

The ASAP Installer checks for a database to connect to during the installation process. Ensure that a database is up and running before you start installing ASAP. If you already have a database up and running, create one or more tablespaces for the ASAP schemas data and index segments.

For information about installing Oracle Database, see the Oracle Database installation documentation.

Oracle recommends that the installation and configuration of Oracle Database be performed by an experienced Oracle Database Administrator (DBA). In addition, some UNIX administrative skills are required to uncompress software, create users and groups, and similar tasks.

Before you start Database client installation, you must install the required Linux packages. . See "Required Linux Packages for ASAP on Linux 8 " for the required Linux 64-bit packages.

Downloading and Installing the Oracle Database Software

Oracle requires that you use the 64-bit Oracle database software. Download and install Oracle Database for this version of ASAP. See *ASAP Compatibility Matrix* for the appropriate version of Oracle Database to install.

For information on installing Oracle Database, see the Oracle Database installation documentation.

Downloading and Installing the Oracle Client Software

You require 64 bit Database client for ASAP. Ensure that the database client, JDBC drivers, and Universal Connection Pool jars are compatible with JDK 17.

See *ASAP Compatibility Matrix* for the appropriate version of Oracle Client software to install.

Installing Base Client

You require Oracle Database Client golden base for ASAP installation.

To install the base client:

1. Download Database Client golden base for Linux 64 bit.
2. For Linux 8, run the following command.

   ```
   export CV_ASSUME_DISTID=OL7
   ```

3. Install the Linux packages using the following command:

```
sudo dnf install bc binutils elfutils-libelf fontconfig glibc glibc-devel ksh
libaio libasan liblsan libX11 libXau libXi libXrender libXtst libgcc
libibverbs libnsl librdmacm libstdc++ libxcb libvirt-libs make policycoreutils
policycoreutils-python-utils smartmontools sysstat
```

4. Unzip the downloaded client file and run the following command:

```
runInstaller
```

   The `runInstaller` command is used to start the Oracle Database Client installation. Respond to the configuration prompts as needed.

   You can ignore the following error by selecting **ignoreAll** in the prompt:

```
Unable to install compat-libcap1-1-10
```

Updating OPatch

OPatch is a Java based utility that enables the application and rollback of patches to Oracle software. It is recommended to use the latest version of OPatch.

To update OPatch:

1. Download the latest OPatch from My Oracle Support.
2. Copy the downloaded file to the database server.
   a. Backup and remove the current OPatch directory in the *Oracle_home* directory.

```
cd $ORACLE_HOME
tar -cvzPpf OPatch_backup.tar OPatch/
```

   b. Remove the OPatch directory after backing up the existing OPatch directory.

```
rm -rf OPatch/
```

   c. Move the downloaded latest zip file to *Oracle_home*.

```
mv <latest version>.zip $ORACLE_HOME
```

   d. Unzip the downloaded OPatch.zip file.

```
cd $ORACLE_HOME
unzip <opatch_latest>.zip
```

3. Run the following command to confirm the OPatch version:

```
$ORACLE_HOME/OPatch/opatch version
```

   A sample response is:

```
OPatch Version:12.2.0.1.24
OPatch succeeded.
```

> **Note:**
>
> You must use the OPatch utility version 12.2.0.1.34 or higher.

Updating Database Client with Release Updates

You must upgrade the client to any version above 19.18.0.0.0 by patching with the appropriate Oracle Database Release Updates (RU).

To update the database client with release updates:

1. Download the appropriate release update from My Oracle Support for the database version that you are using.
2. Copy the zip file to database client host machine.
3. Unzip the downloaded RU file.

   ```
   unzip <latest_version_RU>.zip
   ```

4. Set the environment variables for ORACLE_HOME and the PATH by running the following commands:

   ```
   export ORACLE_HOME=client base path
   export PATH=$ORACLE_HOME/OPatch/opatch:$PATH
   ```

5. Set your current directory to the directory where the RU file is located and use the commands provided in the RU **Readme** file to apply the OPatch utility. For example:

   ```
   cd UNZIPPED_PATCH_LOCATION/36916690/36912597
   opatch apply
   ```

6. Check the database version using the command:

   ```
   $ORACLE_HOME/bin/sqlplus -v
   SQL*Plus: Release 19.0.0.0.0 - Production
   Version 19.18.0.0.0
   ```

Installing, Creating, and Configuring an ASAP Database Instance (Single or Oracle RAC)

This section describes the overall workflow for creating and configuring a single database instance or an Oracle Real Application Clusters (RAC) database instance for use with ASAP.

To install, create, and configure Oracle RAC or Single database instance for ASAP:

1. After installing the Oracle database and client, apply any appropriate software patches.

> **Note:**
>
> See *ASAP Compatibility Matrix* for the required software patches.
>
> Some of the scripts that run with Oracle Database patch installation can take several hours to complete. Plan accordingly, and ensure that all patch scripts run correctly to completion and return to the UNIX prompt.

2. Create a database instance.

> **Note:**
>
> Review and apply recommended database instance initialization parameters listed in "Configuring the Oracle Database Initialization Parameters". Some default database initialization parameters must be configured for ASAP to run.
>
> For an Oracle RAC database, follow the additional database connection recommendations described in "Configuring Oracle RAC Database Connectivity."

3. Ensure that you have determined the database tablespace and Oracle Database Server configuration that you require. See " Configuring the Oracle Database Connection Mode."
4. Start the Oracle Database instance.
5. Create one or more ASAP tablespaces and redo log files. (Oracle recommends that your Oracle DBA create the ASAP tablespaces.)

   For more information, see "Recommended Tablespace and Redo Log Sizes for ASAP" and "Creating Oracle Database Tablespaces, Tablespace User, and Granting Permissions."
6. Switch to an Oracle Database admin account (for example, **SYS**) and run the **ora_system_procs.sh** script to populate the common PL/SQL functions used by multiple instances of ASAP.

> **Note:**
>
> The *ASAP_src***/asap_utils/ora_system_procs.sh** script should not be run more than once against a single Oracle Database server.

7. Verify the Oracle Database installation and configuration.

Configuring the Oracle Database Connection Mode

Configure your Oracle database to accept connections from ASAP using the dedicated connection mode. For each Oracle database connection, a dedicated server process is created which does work on behalf of the ASAP user session. All interactions with the Oracle Shared Global Area (SGA) and data files are done through this dedicated server processes. A server process is started for every user session connecting to the database.

Configuring the Oracle Database Maximum Memory Target

The recommended system size of Oracle Database maximum memory target for the automatic meomry manager (AMM) is 16GB.

Configuring the Oracle Database Initialization Parameters

In Oracle Database instance installations, you can use either **init$ORACLE_SID.ora** or **spfile$ORACLE_SID.ora**.to specify initialization parameters.

The most secure way of modifying Oracle Database parameters, is to specify them on the SQL*PLUS command line, storing them in the spfile. This way, each modification is parsed as you enter it and the parameters are correctly entered.

To show the value of a parameter in the spfile, and then change it, use the following procedure:

1. Start **sqlplus**.
2. Log in using **/as sysdba.**
3. Enter:

```
SHOW PARAMETER PROCESSES
ALTER SYSTEM SET PROCESSES='200' SCOPE=spfile;
```

To show all parameters that start with certain characters, use a common abbreviation. For example, the following command shows all parameters that start with **DB**.

```
SQL> SHOW PARAMETER DB
```

> **Note:**
>
> Some parameters will not take effect until the database instance is stopped, and then restarted. The command startup force can accomplish this.

To manually edit parameters in the **init$ORACLE_SID.ora** file, convert it to an spfile.

> **Note:**
>
> Do not manually edit the **spfile$ORACLE_SID.ora**. Editing a spfile corrupts it. This can prevent you from starting an instance, or cause an active instance to fail.

To convert the INIT.ora file to an spfile:

1. Start **sqlplus**.
2. Log in using **as sysdba** for the user name.

   ```
   sqlplus sys/password as sysdba
   ```

   Where *password* is the password for your **sys** database user account.
3. Enter:

```
create spfile from pfile;
File created.

CREATE SPFILE ['spfile_name'] FROM PFILE ['pfile_name'];
COMMIT;
```

To return parameters to the INIT.ORA type parameter file from an spfile:

1. Start **sqlplus**.
2. Log in using **as sysdba** for the user-name.

```
sqlplus sys/password as sysdba
```

3. Enter:

```
create pfile from spfile;

File created.

CREATE PFILE ['pfile_name'] FROM SPFILE ['spfile_name'];
COMMIT;
```

Recommended Oracle Database Initialization Parameter Settings

This section provides recommended Oracle Database initialization parameter values for use with ASAP in a production system. The suggested values are guidelines only. The optimum values for your system will depend on the particulars of your installation, and your processing requirements.

Table 5-1 lists suggested Oracle Database initialization parameter values.

> **Note:**
>
> You must configure the following parameters. However, the particular values you implement will depend on your chosen configuration.

Table 5-1 Database Initialization (init$ORACLE_SID.ora) Parameters

Parameter and Value	Description
PROCESSES =3000 and 5000	PROCESSES specifies the maximum number of operating system user processes that can simultaneously connect to Oracle Database. Its value should allow for all background processes such as locks, job queue processes, and parallel execution processes. The default values of the SESSIONS and TRANSACTIONS parameters are derived from this parameter. Therefore, if you change the value of PROCESSES, you should evaluate whether to adjust the values of those derived parameters. For more information on Configuration Variables, see the *ASAP System Administrator's Guide*.
SESSIONS = 1600 or up to 600 per online instance (production) 100 - 120 per online instance (development)	This parameter sets the maximum number of connections to the Oracle Database.
COMMIT_WAIT = 'WAIT'	This parameter controls when the redo for a commit is flushed to the redo logs.

Table 5-1 (Cont.) Database Initialization (init$ORACLE_SID.ora) Parameters

Parameter and Value	Description
FILESYSTEMIO_OPTION = 'SETALL'	Specifies I/O operations for file system files.

Creating Oracle Database Tablespaces, Tablespace User, and Granting Permissions

In "ASAP Oracle Database Tablespace Sizing Requirements" you determine how you will map ASAP server schemas to tablespaces, what the tablespaces will be named, and what their size requirements are.

Typically, all environments for testing can share the tablespaces. However, in Oracle Database you can distribute the ASAP components by specifying different default tablespaces for each environment before installing the ASAP database definitions.

Some ASAP implementations may require a tablespace layout that differs from the layout recommended here. The ASAP administrator, in conjunction with the Oracle DBA, can create any tablespace layout that is required.

Oracle recommends that tablespaces be created with the following default properties:

- Size – as determined using the sizing information provided in "Recommended Tablespace and Redo Log Sizes for ASAP."
- Storage Clause:
  - Initial 256 GB
  - Next 256 KB
  - Percent increase 0
  - Maxextents Unlimited

> **Note:**
>
> In development systems, a database can remains idle for longer than the default database connection time-out setting. This can cause the database to terminate the connections to the ASAP servers generating the ORA-02396 event. This scenario can be avoided by setting the database connection time-out parameter to a higher value.
>
> This scenario is unlikely to occur in a production system.

To create the tablespaces, tablespace users, and grant permissions in the Oracle Database:

1. Log in to your Oracle database as a sysdba.
2. Start the Oracle database.
3. Create the tablespaces.

   For example:

```
create tablespace "ASAPDATA_TS" logging datafile '/u01/app/oracle/oradata/ASAP72/
ASAPDATA_TS1.dbf' size 31999M reuse autoextend on next 5120K maxsize unlimited
default storage (initial 256K next 256K minextents 1 maxextents 2147483645
pctincrease 0);

create tablespace "ASAPINDEX_TS" logging datafile '/u01/app/oracle/oradata/ASAP72/
ASAPINDEX_TS1.dbf' size 31999M reuse autoextend on next 5120K maxsize unlimited
default storage (initial 256K next 256K minextents 1 maxextents 2147483645
pctincrease 0);

CREATE USER "ASAP_SYS" PROFILE "DEFAULT" IDENTIFIED BY "ASAP_SYS" DEFAULT TABLESPACE
"ASAPDATA_TS" TEMPORARY TABLESPACE "TEMP" ACCOUNT UNLOCK;

GRANT CREATE ANY CONTEXT TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT CREATE ANY TABLE TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT CREATE ANY VIEW TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT CREATE SNAPSHOT TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT CREATE SYNONYM TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT CREATE TABLE TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT CREATE USER TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT CREATE VIEW TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT GRANT ANY PRIVILEGE TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT QUERY REWRITE TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT UNLIMITED TABLESPACE TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT SELECT ON "SYS"."V_$PARAMETER" TO "ASAP_SYS" WITH GRANT OPTION;
GRANT "CONNECT" TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT "DBA" TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT "EXP_FULL_DATABASE" TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT "IMP_FULL_DATABASE" TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT "RESOURCE" TO "ASAP_SYS" WITH ADMIN OPTION;
GRANT EXECUTE ON SYS.DBMS_LOCK TO "ASAP_SYS" WITH GRANT OPTION;
GRANT SELECT ON DBA_TABLESPACES TO "ASAP_SYS" WITH GRANT OPTION;
```

Creating Oracle Database Users

Each ASAP server schema requires an Oracle user name and password. You can pre-configure these accounts and supply the needed information during the installation; however, this step is optional. The ASAP installation will automatically create the needed users unless you indicate they have already been created.

To create the Oracle Database users using SQL Plus commands:

1. Start SQL Plus and log in as the Oracle admin user.

   ```
   sqlplus -s Oracle_admin/Oracle_admin_pw@Oracle_SID
   ```

   Where *Oracle_admin* and *Oracle_admin_pw* is an Oracle Database administration account and password and *Oracle_SID* is the Oracle Database SID.

2. Create the users. In the example below, you would replace the *ASAP_envID* tags with the actual ASAP environment ID you want to use when you install ASAP, and replace the **_user_password* entities with the actual passwords for the users.

   ```
   create user admASAP_envID identified by adm_user_password default tablespace
   Tablespace_name temporary tablespace $TEMP_TS;
   grant connect, resource, unlimited tablespace to admASAP_envID ;
   create user sarmASAP_envID identified by sarm_user_password default tablespace
   Tablespace_name temporary tablespace $TEMP_TS;
   grant connect, resource, unlimited tablespace to sarmASAP_envID ;
   create user nepASAP_envID identified by nep_user_password default tablespace
   Tablespace_name temporary tablespace $TEMP_TS ;
   grant connect, resource, unlimited tablespace to nepASAP_envID;
   ```

```
create user ctrlASAP_envID identified by ctrl_user_password default tablespace
Tablespace_name temporary tablespace $TEMP_TS;
grant connect, resource, unlimited tablespace to ctrlASAP_envID;
commit;
```

3. Record the actual names used in the table.

To create the Oracle Database users using the ASAP sample create users script:

1. Download the ASAP installation software as described in "Downloading the ASAP Installation Files."
2. Using a text editor, open *ASAP_src***/asap_utils/sample-create-user.ksh**.
3. Edit the following variables:
   - ENV_ID=*env_id*: Where *env_id* is the environment ID that you will use for your ASAP installation.
   - USER=*db_user*: Where *db_user* is database user with sufficient privileges to create new tablespace users.
   - PASSWORD=*db_pw*: Where *db_pw* is database password for the database user with sufficient privileges to create new tablespace users.
   - ORACLE_SID=*sid*: Where *sid* is the database SID value.
   - TEMP_TS=*temp_ts*: Where *temp_ts* is the temporary tablespace associated to each user.
4. Run the *ASAP_src***/asap_utils/sample-create-user.ksh** script.

If you use different user names, ensure that you supply the same names during the ASAP server installation.

Patch the Oracle Database Server Software

You may need to apply a patch to the Oracle Database Server software after installation. For details, see *ASAP Compatibility Matrix*.

Configuring Oracle RAC Database Connectivity

ASAP uses the availability features of the Oracle RAC system. ASAP supports an active/passive failover Oracle RAC database configuration. If you have installed ASAP in an Oracle RAC system, ASAP automatically connects to the active secondary database node when the primary database node fails over to the secondary. ASAP Control Server remains running during the Oracle RAC node switchover. If a transaction is in progress when the Oracle RAC node fails over and, as a result, if any ASAP server shuts down, ASAP Control Server automatically restarts the failed ASAP processes.

Follow the recommendations in the Oracle RAC documentation and use cluster node Virtual IP Addresses (VIP) when configuring Oracle LOCAL listeners (*LOCAL_LISTENER*). For more information on listener configuration and Oracle RAC installation, refer to the **Oracle Real Application Clusters Installation Guide** and the **Oracle Database Net Services Administrator's Guide**.

To configure the **TNSNAMES.ora** file for Oracle RAC support:

1. Using a text editor, open **TNSNAMES.ora** file in the *Oracle_home***/network/admin/** directory.
2. Configure the **TNSNAMES.ora** file entries listed in Table 5-2.

Table 5-2 TNSNAMES.ora Entries

Entry	Description
LOAD_BALANCE	This parameter specifies whether load balancing is enabled between Oracle RAC databases. This parameter is embedded in the ADDRESS_LIST parameter in the **tnsnames.ora** file. Set this attribute to **NO**. ASAP does not support active/active RAC databases configuration.
FAILOVER	This parameter specifies whether connect-time failover is enabled between primary and secondary Oracle RAC databases. This parameter is embedded in the ADDRESS_LIST parameter in the tnsnames.ora file Set this attribute to TRUE. ASAP supports the active/passive Oracle RAC database configurations.
ADDRESS	This parameter contains the address information for an Oracle RAC database. Include two instances of this attribute for the primary and secondary Oracle RAC databases. This parameter is embedded in the ADDRESS_LIST parameter in the tnsnames.ora file The ADDRESS parameters includes the following parameters: • PROTOCOL: Specifies the protocol used to connect to the Oracle RAC database. For example, TCP. • HOST: Specifies the host name or IP address for the RAC database. • PORT: Specifies the port number for the Oracle RAC database.
SERVER	This parameter directs the listener to connect the client to a specific service handler. Set this parameter to DEDICATED.
SERVICE_NAME	This parameter identifies the Oracle Database database service to access. Set the value to a value specified by the SERVICE_NAMES parameter in the initialization parameter file.

Table 5-2 (Cont.) TNSNAMES.ora Entries

Entry	Description
FAILOVER_MODE	This parameter contains the connection information required for an Oracle RAC active/passive failover database configuration and transparent application failover (TAF). It instructs Oracle Net to fail over to a different listener if the first listener fails during run time. Do not confused this parameter with the connect-time failover FAILOVER parameter. This parameter is embedded in the CONNECT_DATA parameter in the tnsnames.ora file. The FAILOVER_MODE parameter includes the following parameters: • TYPE: Set `TYPE` to `session`. `TYPE=session` establishes a new connection to the Oracle active instance, and any work in progress is lost. ASAP will replay all such failed transactions related to the Control Server. • METHOD: Specify the method to be used for establishing the connection. `METHOD=preconnect` directs TAF to create two connections at transaction startup time: one to the primary database and a backup connection to the backup database. `METHOD=basic` directs TAF to create a connection to the primary database at transaction startup time. Connection to the backup database is established only after the connection to the primary database fails. • RETRIES: Specify the number of times you want ASAP to retry the connection when the connection to the database fails. For example, `RETRIES=11`. • DELAY: Specify the interval in seconds between connection retries. For example, `DELAY= 6`.

For example:

```
DB_CONNECT=
   (DESCRIPTION =
         (LOAD_BALANCE=NO)
         (FAILOVER=TRUE)
         (ADDRESS =
            (PROTOCOL = TCP)
            (HOST = test1.system.com)
            (PORT = 1521)
         )
         (ADDRESS =
            (PROTOCOL = TCP)
            (HOST = test2.system.com)
            (PORT = 1521)
         )
      )
      (CONNECT_DATA =
         (SERVER = DEDICATED)
         (SERVICE_NAME = TEST)
         (FAILOVER_MODE=
            (TYPE=select)
            (METHOD=basic)
            (RETRIES=30)
            (DELAY=15)
         )
```

```
        )
    )
```

3. Install ASAP.
4. Perform the post-installation tasks described in "Configuring ASAP to Support TAF for Oracle RAC."

6

Installing and Configuring Oracle WebLogic Server

This chapter describes the tasks relating to Oracle WebLogic Server that you perform before installing Oracle Communications ASAP.

About Configuring Oracle WebLogic Server

This chapter describes procedures relating to installing the WebLogic Server and other required applications, and also configuring the WebLogic Server domain where you will install ASAP.

See "Required Linux Packages for ASAP on Linux 8" for details on installing required Linux 64-bit packages.

> **Note:**
>
> Ensure that the WebLogic server is running before you install ASAP.

WebLogic Server Java Requirements

ASAP uses WebLogic Server to support Java Platform, Enterprise Edition (Java EE) applications such as the Java SRP, User Security Management, and the Service Activation Deployment Tool (SADT).

See *ASAP Compatibility Matrix* for the required WebLogic Server version, Java Standard Edition (Java SE) Java Development Kit (JDK), and information on required patches for WebLogic Server.

Downloading and Installing the WebLogic Server Software

You must create a WebLogic Server UNIX user account and password to install and configure the WebLogic Server. Make certain that you use this account for any subsequent changes to the WebLogic Server configuration.

If you have an existing WebLogic Server installation, ensure that the versions and patches of your existing installation meet the requirements for ASAP (see *ASAP Compatibility Matrix*).

If you do not have an existing WebLogic Server installation, you must perform a full installation and create a WebLogic Server Domain.

To download and install Oracle WebLogic Server software:

1. Install the appropriate JDK for your platform and the required version of the Oracle WebLogic Server (see "WebLogic Server Java Requirements").

> **Note:**
>
> Ensure that you follow the latest Oracle WebLogic Server installation documentation carefully, performing all required steps.

2. Locate the generic Oracle WebLogic Server software distribution file included with the ASAP software on the Oracle software delivery website.
3. Download the generic Oracle WebLogic Server software distribution file for your hardware platform. (You may have to save it to an interim machine, and then use FTP to transfer the file to your target machine.)
4. Switch to the UNIX account you created to install the WebLogic Server:

   ```
   su - WLS_user
   ```

   Where *WLS_user* is the name of the account you used to install the WebLogic Server.
5. Change the permissions on the installation JAR file so that your WebLogic Server user can run it. For example:

   ```
   chmod a+x fmw_12.2.1.4_wls.jar
   ```
6. If you are running X-Windows on a remote display, export the display. For example:

   ```
   export DISPLAY=10.15.16.17:0.0.
   ```
7. Start the Oracle WebLogic installer. For example:

   ```
   java -jar fmw_12.2.1.4_wls.jar
   ```

> **Note:**
>
> Specify a different directory from the directory containing the install JAR file for the Oracle installation.

8. When the installation is complete, perform domain creation and configuration for ASAP as described in "Creating a WebLogic Server Domain."

Creating a WebLogic Server Domain

The instructions below describe a scenario to configure an admin server domain or an Admin server with managed server domain. You may decide to vary your implementation based on the requirements of your installation. Refer to the Oracle WebLogic Server documentation for details.

> **Note:**
>
> When the Oracle WebLogic Server is started or stopped, you should not have the ASAP Environment_Profile sourced. Some settings in the ASAP CLASSPATH can interfere with the Java libraries needed for log4j and cause errors such as *java.lang.NoClassDefFoundError*. You can use the command:
>
> ```
> unset CLASSPATH
> ```
>
> prior to starting the WebLogic Server to avoid the conflict.

To install and configure an Oracle WebLogic Server domain:

> **Note:**
>
> The paths in the steps below are specified for a standard Oracle WebLogic and Oracle Database configuration. Ensure you use paths appropriate to your installation.
>
> Navigate to *WebLogic_home***/common/bin** and run the following commands at the prompt, substituting the placeholder values with the actual name values:
>
> - export WL_HOME=*domain_home*
> - export ORACLE_HOME=*Oracle_home*
> - export PATH="$PATH:$ORACLE_HOME/bin"

1. From *WebLogic_home***/common/bin**, start the **WebLogic Configuration Wizard** using the following command:

   ```
   ./config.sh
   ```

   The **Configuration Type** screen displays on which you can choose either to Create or Extend a Configuration.
2. On the **Configuration Type** screen, select **Create a new domain**, and in **Domain Location**, specify the location where you want to install your domain.
3. Click **Next**.

   The **Templates** screen is displayed.
4. On the **Templates** screen, select **Create Domain Using Product Templates:**

   > **Note:**
   >
   > **Basic WebLogic Server Domain...** is automatically selected.

5. Click **Next**.

   The **Administrator Account** screen is displayed.
6. On the **Administrator Account** screen, specify the admin user name and password for your WebLogic domain.
7. Click **Next**.

The **Domain Mode and JDK** screen is displayed.

8. On the **Domain Mode and JDK** screen choose from **Development mode** or **Production mode**. Select the location of the installed JDK for WebLogic to use.
9. Click **Next**.

   The **Advanced Configuration** screen is displayed.
10. On the **Advanced Configuration** screen, select the **Administration Server** and **Managed Servers, Clusters and Coherence** check boxes.

> **Note:**
>
> ASAP does not support clusters, but the screen to configure clusters appears after the **Managed Servers** configuration screen. Ignore the **Clusters** screen.

11. Click **Next**.

    The **Administration Server** screen is displayed.
12. On the **Administration Server** screen, specify the values described in Table 6-1.

Table 6-1 Configure the Administration Server screen

Administration Server fields	Values
Name	Enter a name for your WebLogic server.
Listen address	Select **All Local Addresses** from the list.
Listen port	Enter a port number for your WebLogic server.
SSL listen port	Enter an SSL port number for your WebLogic server. The **SSL enabled** check box must be selected before a value can be entered for this field.
SSL enabled	Select this check box to enable SSL.

13. Click **Next**.

    The **Managed Servers** screen is displayed.
14. On the **Managed Servers** screen, enter values for the managed server that you want to configure. To configure a managed server click **Add**. The **Listen address** should match the IP address of the WebLogic Server host. Enable SSL and specify a listen and an SSL port.
15. After you add your managed server specified, click **Next**.

    The **Clusters** screen is displayed.
16. Click **Next**.

    The **Machines** screen is displayed.
17. Select the **Unix Machine** tab and add and configure the machines that you want. For more information about configuration options, see WebLogic Server documentation.
18. Click **Next**.

    The **Assign Servers to Machines** screen is displayed.
19. After you assign servers to machines, click **Next**.

    The **Configuration Summary** screen is displayed.

20. Review the configuration settings you have made. Click **Create**.

    The **Configuration Progress** screen is displayed.

21. When creation is 100% complete, click **Next**.

    The **Configuration Success** screen is displayed.

22. Click **Finish**.

    For more information on the steps required to configure each of these options, see *Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard* for your version of WebLogic Server.

    The main Oracle WebLogic Server software installation is complete – continue with the installation of the required patches below.

Patching the WebLogic Server Software

You may need to apply a patch to the WebLogic Server software after installation.

See *ASAP Compatibility Matrix* for details.

7

Installing ASAP

This chapter describes how to install Oracle Communications ASAP. Before installing ASAP, read these chapters:

- ASAP Installation Overview
- Planning Your ASAP Installation
- ASAP System Requirements
- Configuring the Operating System
- Installing and Configuring Oracle Database and Client
- Installing and Configuring Oracle WebLogic Server

About Installing the ASAP Software

> **Note:**
>
> The term *ASAP Server* is used throughout this chapter when addressing all ASAP servers collectively.

Oracle recommends that you back up the Oracle WebLogic Server core software and the WebLogic Server domain before installing ASAP. For installation, ASAP must be linked to an as-yet unused WebLogic Server domain. If the ASAP installation fails, and you have not backed up the WebLogic Server domain information, you must recreate the WebLogic Server domain before installing ASAP. See "Creating a Backup of the WebLogic Server."

> **Note:**
>
> Before installing a new version of ASAP, Oracle recommends that you back up all data.

Prerequisites for Installing ASAP

The prerequisites for installing ASAP are:

- **Weblogic Server**: Ensure the WebLogic Server domain has been created and the required server instance has been configured.
- **Database Planning**: Ensure the required Database client is installed on the Linux VM and also that the Database is up and running.
- **Java**: Ensure that you are using Java version 17. Refer to the *ASAP Compatibility Matrix* for specific minimum update requirements. Also, ensure that you have set the **JAVA_HOME_ASAP** environment variable.

- **Apache Ant**: Ensure that the required version of Ant is installed.

Downloading the ASAP Installation Files

ASAP installation package is offered as a RedHat Package Manager (RPM) package. On Linux systems, you install the ASAP RPM package using the RPM package manager or the DNF.

Leveraging DNF, you can install the ASAP Installer on your Linux machines. DNF installation offers comprehensive traceability, enabling you to track the installation of different ASAP versions and their respective installation locations.

Regardless of the chosen tool, it is recommended that you override the default installation location. This ensures that you can easily access patches and version upgrades in the future. Additionally, overriding the installation location facilitates upgrade preparation and rollback scenarios.

To download the ASAP package installer:

1. Download the ASAP software for your operating system from the Oracle software delivery website, located at:

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

2. Save the downloaded file (for example, **ASAP.R7_4_1.B*xxx*.linux.tar**) to a temporary directory.

Configuring, Backing Up, and Starting WebLogic Before Installing ASAP

To configure, backup, and start the WebLogic server, use the following procedure:

> **Note:**
>
> When you start your WebLogic Server instance, make sure that the *ASAP_home*/**Environment_Profile** has not been sourced in your WebLogic server terminal session. This file references environment variables that are incompatible with your WebLogic server. In the context of this procedure, ASAP has not been installed, and this scenario is not possible; however, it can occur after you have installed ASAP.

1. Archive your WebLogic Server domain. For example,

   ```
   cd domain_home/
   tar cf domain_name.tar domain_home/
   ```

   Where *domain_name* is the name you want to give your WebLogic Server domain backup file.

2. Start the WebLogic server. You can use **nohup** to run your WebLogic server in the background.

   ```
   cd domain_home/
   nohup ./startWebLogic.sh &
   ```

3. Monitor the **nohup.out** file to make sure the domain goes into the RUNNING mode.

   ```
   tail -f nohup.out
   ```

   The output from this command should end with the following text when it enters the **RUNNING** mode.

   ```
   <Jul 15, 2014 2:50:29 PM EDT> <Notice> <WebLogicServer> <BEA-000360> <Server started
   in RUNNING mode>
   ```

4. Stop the **tail** process.

   ```
   Ctrl-C
   ```

5. Exit from your oracle user session.

   ```
   exit
   ```

Re-Running the ASAP Server Installation

> **Note:**
>
> Oracle recommends a new complete installation in case of a failed installation. Re-running a failed installation will result in another failed installation.

To re-run the ASAP Server installation:

1. Stop the WebLogic Server:

   ```
   cd domain_home/bin
   ./stopWebLogic.sh -U wls_admin_user
   ```

2. Remove the old WebLogic domain.
3. Drop the Database as appropriate.
4. Remove the installed RPM package by running the following commands:

   ```
   #Get the name of the rpm
   rpm -q asap-installer
   #Remove the RPM package
   rpm -ev rpm_name
   ```

5. Remove all the directories and files that were created by ASAP installation by running the following command:

   ```
   rm -rf
   ```

6. Perform a fresh complete installation of ASAP. See Installing an ASAP Demonstration System for the steps involved in performing a fresh complete installation.

Installing an ASAP Demonstration System

The instructions in this section assume that you are installing all of the ASAP components, including the database, on a single machine. This machine should meet the hardware requirements for Oracle Linux.

The procedure for installing a demonstration system include:

1. Installing Oracle Linux with the Oracle Validated Package
2. Preparing the Linux Environment
3. Downloading and Extracting the ASAP Installation Files
4. Downloading and Installing the Oracle Database and Client Software
5. Configuring Database Listener
6. Creating the Database Instance
7. Creating Tablespace, Tablespace Users, and Granting User Privileges
8. Downloading and Installing the WebLogic Server Software
9. Creating and Configuring a WebLogic Server Domain
10. Installing ASAP
11. Configuring ASAP

Installing Oracle Linux with the Oracle Validated Package

Use the following procedure to install Oracle Linux with the Oracle validated software package.

1. On a Windows workstation, launch Internet Explorer and log in to the Oracle software delivery website.
2. Select and download the Oracle Linux version required for the x86 64 bit platform (see *ASAP Compatibility Matrix*).
3. Install Oracle Linux on an x86 64 bit platform.
4. Select default installation options until you reach the page where you get the option to customize your installation. Select the **Customize now**, and click **Next**.
5. Select **Base System** from the left menu.
6. Select **System Tools** from the right menu.
7. Click **Optional packages**.
8. Select the Oracle Validated RPM package from the package list, and click **Next**.

   This package provides the following features:

   - Preconfigured Oracle Linux kernel configuration, suitable for the Oracle database. These kernel configuration attributes are sufficient for an ASAP demonstration system; however, for a production environment or test system, configure your kernel parameters as described in "Setting Linux Kernel Parameters."
   - All Oracle packages required to install and run the Oracle database software.
   - Preconfigured Oracle database user and groups for database installation.
9. Click **Close**.
10. Click **Next** and continue the installation process.
11. During the Oracle Linux configuration process, select the following options:
    - Enable the firewall and set **SSH**, **Telnet**, and **FTP** as trusted protocols.
    - Set SELinux to **Permissive**.
    - Check **Enable Network Time Protocol** in the Network Time Protocol tab.

Preparing the Linux Environment

Use the following procedure to prepare your Oracle Linux environment. The steps described in this procedure include:

- Installing the JDK for WebLogic Server
- Creating a symbolic link to the JDK from Mozilla Firefox to enable the Java plugin
- Creating an ASAP user and group
- Configuring a host name for your installation in the **/etc/hosts** file
- Creating the directories for your Oracle Database installation
- Configuring the Oracle user **.bashrc** file

To prepare the Linux environment:

1. Log in to Oracle Linux as **root**.
2. For more information about the compatible versions, refer to the *ASAP Compatibility Matrix*.
3. Change the permissions on the Java installation file:

   ```
   chmod a+x jdk_install_file
   ```

   Where *jdk_install_file* is the Java installation file.
4. Move the file to a system-wide location such as **/usr/local**. For example:

   ```
   mv jdk_install_file /usr/local
   ```
5. From the new location, install Java SE.

   ```
   ./jdk_install_file
   ```
6. Create a symbolic link from the **/usr/lib/mozilla/plugins** directory to *Java_home* **/jre/lib/i386/libnpjp2.so** file (where *Java_home* is the location where you installed the JDK) to enable the Java plugin for Mozilla Firefox. For example:

   ```
   cd /usr/lib/mozilla/plugins
   ln -s /usr/local/jdk1.8.x/jre/lib/i386/libnpjp2.so
   ```
7. Create an ASAP user and group as described in "Configuring a UNIX ASAP Group and User."
8. Determine the IP address for your platform.

   ```
   ifconfig
   ```
9. Edit the **/etc/hosts** file and provide a host name for your IP address. For example:

   ```
   # Do not remove the following line, or various programs
   # that require network functionality will fail.
   127.0.0.1 localhost.localdomain localhost loopback
   ::1 localhost6.localdomain6 localhost6
   10.156.53.74 testsystem.oracle.us.com testsystem loghost
   ```
10. Create and assign permissions to folders in preparation to install your Oracle database. For example:

    ```
    mkdir -p /u01/app/
    chown -R oracle:oinstall /u01/app/
    chmod -R 775 /u01/app/
    mkdir /u01/app/oradata /u01/app/fast_recovery_area
    ```

```
chown oracle:oinstall /u01/app/oradata/u01/app/fast_recovery_area
chmod 755 /u01/app/oradata /u01/app/fast_recovery_area
```

11. Edit the Oracle user's **.bashrc** system setup file using a text editor such as **gedit** or **vi**. This example provides common Oracle database environment variables and paths:

```
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
. /etc/bashrc
fi

# User specific aliases and functions
export ORACLE_SID=ASAP
export ORACLE_HOME=/u01/app/oracle/product/12.1.3/dbhome_1
PATH=$PATH:$HOME/bin:/u01/app/oracle/product/12.1.3/dbhome_1/bin/
export PATH

umask 022
```

Downloading and Extracting the ASAP Installation Files

To download the ASAP installation files use the procedure described in "Downloading the ASAP Installation Files" with the following exceptions:

- Log in to your Oracle Linux workstation with your **asap** user account.
- Download the file directly to your Oracle Linux workstation.
- For this demonstration system, create a new directory called ***asapinstaller*** in the **/home/asap** directory and extract the ASAP installation files in this directory. For example:

```
cd /home/asap
mkdir asapinstaller
mv ASAP.R7_4_1.Bxxx.linux.tar asapinstaller/
cd asapinstaller
tar -xvf ASAP.R7_4_1.Bxxx.linux.tar
```

Downloading and Installing the Oracle Database and Client Software

Use the following procedure to download and install the Oracle database and client software.

1. Log in as the Oracle user.
2. Log in to the Oracle software delivery website.
3. Select and download the Oracle Database version you require for the Linux x86-64 platform. For more information about the compatible versions, refer to the *ASAP Compatibility Matrix*.

> **Note:**
>
> Do not download the client from the Oracle software delivery website.

4. Use the following command to extract the content from the installation files:

```
unzip file_name.zip
```

Where *file_name* is the name of the database installation file. There are typically two installation files.

5. Run the database installer.

```
./database/runInstaller
```

6. During the installation process select the default options for all fields except for the following:
   a. Install database software only
   b. Single Database Instance
   c. Enterprise Edition
7. Follow the GUI instructions to finish the installation.
8. Download and install the client as described in "Downloading and Installing the Oracle Client Software" using the default options for all fields except for the following:
   a. Select the **Administrator** Installation Type
   b. Enter **/u01/app/oracle/product/19.3.0.0/client_1** for the **Software Location**
   c. Enterprise Edition

Configuring Database Listener

Use the following procedure to configure a database listener.

1. Run the Network Configuration Assistant (**netca**).

```
netca
```

   The **Network Configuration Assistant: Welcome** screen appears.
2. Select **Listener Configuration**.

   The **Network Configuration Assistant: Listener Configuration Listener** screen appears.
3. Select **Add**.

   The **Network Configuration Assistant: Listener Configuration, Listener Name** screen appears.
4. Accept default **LISTENER**. Click **Next**.

   The **Network Configuration Assistant: Listener Configuration, Select Protocols** screen appears.
5. Select **Next** to accept the default TCP selection.

   The **Network Configuration Assistant: Listener Configuration, TCP/IP Protocol** screen appears.
6. Select **Use the standard port number 1521**. Click **Next**.

   The **Network Configuration Assistant: Listener Configuration, More Listeners?** screen appears.
7. Select **No**. Click **Next**.

   The **Network Configuration Assistant: Listener Configuration Done** screen appears.
8. Click **Next**.
9. Click **Finish**.

Creating the Database Instance

Use the following procedure to configure a database instance.

1. Run the Database Configuration Assistant (**dbca**).

   ```
   dbca
   ```

   The **Database Configuration Assistant: Welcome** screen appears.
2. Click **Next**.

   The **Database Configuration Assistant: Step 1 of 12: Operations** screen appears.
3. Select **Create a Database**.
4. Click **Next**.

   The **Database Configuration Assistant: Step 2 of 12: Database Templates** screen appears.
5. Select **General Purpose or Transaction Processing**.
6. Click **Next**.

   The **Database Configuration Assistant: Step 3 of 12: Database Identification** screen appears.
7. Enter **ASAP** for the **Global Database Name** and **SID** fields.
8. Click **Next**.

   The **Database Configuration Assistant: Step 4 of 12: Management Options** screen appears.
9. Click **Next** to accept the default options.

   The **Database Configuration Assistant: Step 5 of 12: Database Credentials** screen appears.
10. Select **Use the Same Administrative Password for All Accounts**.
11. Enter a **Password** in the **Password** and **Confirm Password** fields.

    > **Note:**
    >
    > Mark down the password you used. You will need this password for the procedure described in "Creating Tablespace, Tablespace Users, and Granting User Privileges."

12. Click **Next**.

    The **Database Configuration Assistant: Step 6 of 12: Database File Locations** screen appears.
13. Click **Next** to accept the default options for storage type and location of data files.

    The **Database Configuration Assistant: Step 7 of 12: Recovery Configuration** screen appears.
14. Click **Next** to accept the default flash recovery area options.

    The **Database Configuration Assistant: Step 8 of 12: Database Content** screen appears.

15. Click **Next** to accept default options for **Sample Schema** and **Custom Scripts** tabs.

    The **Database Configuration Assistant: Step 9 of 12: Initialization Parameters** screen appears.

16. Click **Finish**.

    The **Create Database - Summary** screen appears.

17. Click **OK** to create your database.

18. Click **Exit**.

Creating Tablespace, Tablespace Users, and Granting User Privileges

Use the following procedure to create a tablespace, tablespace user, and grant privileges.

1. From the Unix terminal, log in to the database instance with the **sys** user account.

   ```
   sqlplus sys/password as sysdba
   ```

   Where *password* is the password you choose for your **sys** account (see "Creating the Database Instance").

2. Run the following command to create your tablespace.

   ```
   CREATE TABLESPACE "POOL_TS" LOGGING DATAFILE '/u01/app/oracle/POOLS_TS.dbf' SIZE
   6000M REUSE AUTOEXTEND ON NEXT 1000K MAXSIZE UNLIMITED DEFAULT STORAGE (INITIAL 128K
   NEXT 128K MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0);
   ```

3. Create a new user for the tablespace. For example:

   ```
   CREATE USER "ASAP_SYS" PROFILE "DEFAULT" IDENTIFIED BY "ASAP_SYS" DEFAULT TABLESPACE
   "POOL_TS" TEMPORARY TABLESPACE "TEMP" ACCOUNT UNLOCK;
   ```

4. Grant privileges to your tablespace user account. For example:

   ```
   GRANT CREATE ANY CONTEXT TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT CREATE ANY TABLE TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT CREATE ANY VIEW TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT CREATE SNAPSHOT TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT CREATE SYNONYM TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT CREATE TABLE TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT CREATE USER TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT CREATE VIEW TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT GRANT ANY PRIVILEGE TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT QUERY REWRITE TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT UNLIMITED TABLESPACE TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT SELECT ON "SYS"."V_$PARAMETER" TO "ASAP_SYS" WITH GRANT OPTION;
   GRANT "CONNECT" TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT "DBA" TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT "EXP_FULL_DATABASE" TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT "IMP_FULL_DATABASE" TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT "RESOURCE" TO "ASAP_SYS" WITH ADMIN OPTION;
   GRANT EXECUTE ON SYS.DBMS_LOCK TO "ASAP_SYS" WITH GRANT OPTION;
   GRANT SELECT ON DBA_TABLESPACES TO "ASAP_SYS" WITH GRANT OPTION;
   ```

5. Log out of sqlplus.

   ```
   exit
   ```

6. Log in to the root user account from the Oracle user session.

   ```
   su root
   ```

7. Copy the **ora_system_procs.sh** from the ASAP installation directory to the Oracle user account home directory.

```
cd /home/asap/asapinstaller/asap_utils/
cp ora_system_procs.sh /home/oracle
```

8. Set permissions for the **ora_system_procs.sh** file.

```
cd /home/oracle
chmod 755 ora_system_procs.sh
```

9. Log out of the root user session.

```
exit
```

10. As the Oracle user, run the **ora_system_procs.sh** script.

```
cd /home/oracle
./ora_system_procs.sh
```

11. Enter **Y** and press **Enter** to continue.
12. Enter the password you created for your **sys** database user account (see "Creating the Database Instance") and press **Enter**.

> **Note:**
>
> The script may produce the following error message.
>
> ```
> drop public synonym set_session
> *
> ERROR at line 1:
> ORA-01432: public synonym to be dropped does not exist
> ```
>
> Ignore this message.

13. Become root user.

```
su root
```

14. Copy the Database **tnsnames.ora** file to your **asap** user account home directory.

```
cp /u01/app/oracle/product/19.3.0/dbhome_1/network/admin/tnsnames.ora /home/asap
```

15. Change the permissions on the **tnsnames.ora** file.

```
cd /home/asap
chmod 755 tnsnames.ora
```

16. Log out of root user.

```
exit
```

17. Log out of Oracle user.

```
exit
```

Downloading and Installing the WebLogic Server Software

Use the following procedure to download and install the WebLogic Server software.

1. Log in as the **asap** user.
2. Log in to the Oracle software delivery website.
3. Select and download the WebLogic version you require for the Linux x86-64 platform. For more information about the compatible versions, refer to the *ASAP Compatibility Matrix*.

4. Change the permissions on the **installation.jar** file so that your WebLogic Server user can run it. For example:

```
chmod a+x fmw_12.2.1.4_wls_lite_Disk1_1of1.zip
```

5. If you are running X-Windows on a remote display, export the display. For example:

```
export DISPLAY=10.15.16.17:0.0.
```

6. Start the WebLogic installer. For example:

```
cd /usr/local/jdk1.8.x/bin/
./java -jar /home/oracle/Desktop/fmw_12.2.1.4_wls_lite_Disk1_1of1.zip
```

> **Note:**
>
> Make sure you use ./ before the Java command to use the correct JDK.

7. Click **Next**.
8. Click **Next** to accept the default Middleware home directory (for example, **/home/asap/Oracle/Middleware**) and follow the installation instructions.
9. Select the **Custom** installation type from the Choose **Install Type** screen.
10. Click **Next**.
11. Deselect the **Coherence Server** component from the component list.
12. Click **Next**.
13. Click **Next** to select the **Local JDK** (for example **Sun SDK 1.8.x (/usr/local/jdk1.8.x)**).
14. Click **Next** to select the default product installation directory for your WebLogic Server (for example **/home/asap/Oracle/Middleware/wlserver_12.2**).
15. Review the Installation Summary, and click **Next** to start the installation process.
16. After the installation is complete, deselect **Run Quickstart**.
17. Click **Done**.

Creating and Configuring a WebLogic Server Domain

Use the following procedure to create and configure a WebLogic Server domain.

1. Log in as the **asap** user.
2. Follow the procedure described in "Creating a WebLogic Server Domain" and select the default options except for those described in Table 7-1.

Table 7-1 WebLogic Server Domain Demonstration System Installation Values

Installer Screen	Fields	Value
Specify Domain Name and Location	**Domain name:** **Domain location:**	**basedomain** **/home/asap/Oracle/Middleware/user_projects/domains**
Configure Administrator User Name and Password	**Name:** **User password:** **Confirm user password:**	**weblogic** ***password*** ***password***

Table 7-1 (Cont.) WebLogic Server Domain Demonstration System Installation Values

Installer Screen	Fields	Value
Select Optional Configuration	Select only the **Administrator Server** option. This demonstration system does not require a managed server.	**N/A**
Configure the Administration Server	**Name:** **Listen address:** **Listen port:** **SSL listen port**: Select **SSL enabled**.	**AdminServer** **All Local Addresses** **7001** **7002**

3. Configure, backup, and start the WebLogic server as described in "Configuring, Backing Up, and Starting WebLogic Before Installing ASAP."

Installing ASAP

You can install the ASAP Installer RPM package on Linux as an RPM or DNF file. RPM is a widely used utility for software installation on Linux systems. ASAP installer is distributed as an RPM relocatable package for Linux 64 bit architectures. The package will install all the required ASAP core files to the install location. The default install location is **/usr/local/asap**. You can also specify a custom install location using `--prefix` option.

Install Using RPM

To install ASAP using RPM:

1. Login to the host machine where you want to install ASAP.
2. Navigate to the directory where the ASAP installation package (**ASAP.R7_4_1.B*xxx*.linux.tar**) is extracted.
3. Run the following command to install ASAP:
   - To install in the custom path:

     ```
     sudo rpm -ivh --prefix=/path/to/desired/installationdir /path/to/rpm/
     asap-installer-7.4.1-Bxxx.x86_64.rpm
     ```

   - To install in the default path (**/usr/local/asap**):

     ```
     sudo rpm -ivh /path/to/rpm/asap-installer-7.4.1-Bxxx.x86_64.rpm
     ```

   ASAP files are installed with the *user:group* that is used to log in to the machine.
4. Run the following command to set the environment variable TNS_ADMIN:

   ```
   export TNS_ADMIN=/path/tnsnames.ora/file
   ```

5. Set the JAVA_HOME_ASAP environment variable by using the following command:

   ```
   export JAVA_HOME_ASAP=/path/to/Java17
   ```

6. The installer logs are found in *ASAP_home*/installer_logs. The database configuration logs are found in the *ASAP_home* directory.

Install Using DNF

When attempting to install a new version of the ASAP installer RPM package, DNF automatically handles the process of either downgrading or upgrading to the specific version. For example:

- If ASAP installer build version B2566 is already installed and an older version, say version B2559 is being installed, DNF will replace or downgrade the existing version B2566 to version B2559.
- Conversely, if build version B2559 is already installed and a newer version, say B2577 is being installed, DNF will replace or upgrade the existing version B2559 to version B2577.

To accommodate multiple versions of the ASAP installer, `installroot` option can be utilized with the `dnf install` command. This allows specifying a custom directory location. For example:

```
#install the RPM package using the dnf command
$ sudo dnf install --installroot=/path/to/desired/installationdir/ /
path/to/rpm/asap-installer-7.4.1-Bxxx.x86_64.rpm
```

> **Note:**
>
> Using DNF incurs a disk cost, as a new root file system is created for each custom location. Approximately 2.5 GB of disk space is required per location when using DNF.

Configuring ASAP

You can configure ASAP using:

- Silent Mode
- Console Mode

Configuring ASAP Using Silent Mode

In Silent mode, you can configure ASAP automatically by providing all the parameters in a configuration file. A sample can be found on **ASAP_home**/config directory as **sampleASAPconfiguration.properties** file. You can rename the file and provide all the configuration parameters. When the configureASAP script is run, the parameters are read from the configuration file provided as input to the script and set up accordingly.

To automatically configure ASAP in the silent mode, run the configureASAP script located at ASAP_home/scripts directory:

```
configureASAP -properties /path/to/configuration/properties -d
```

where -d indicates the demo or silent mode. The values for all the parameters are read from the properties file, provided as input, to the configureASAP script in above command. See "Sample Configuration File" for the details on the parameters in the sample configuration file.

Configuring ASAP Using Console Mode

You can configure ASAP in the console mode by running the following command:

```
configureASAP -properties /path/to/configuration/properties
```

In the console mode, you must configure passwords in the command line when prompted for and rest of the parameters are read from the properties file. A sample can be found on **ASAP_home**/config directory as **sampleASAPconfiguration.properties** file. You can rename the file and provide all the configuration parameters. When configureASAP script is run, the parameters are read from the configuration file, provided as input to the script and set up accordingly. Oracle recommends that you use console mode for configuring ASAP. See "Sample Configuration File" for the details on the parameters in the sample configuration file.

Troubleshooting a Failed ASAP Installation

This section explains how to troubleshoot error messages that you may encounter during an unsuccessful ASAP installation.

Java Heap Space Exception Prevents ASAP Installation

A Java heap exception may be caused because the physical memory, CPU speed, or the quantity of CPUs is less than the minimum requirements. The JVM, which is initiated by the ASAP installer to deploy the EAR file, runs out of memory and returns the Java heap space exception in the log.

Ensure that your system meets or exceeds the minimum requirements, as outlined in "ASAP Server Hardware Requirements."

If you discovers a Java heap space exception in the installation log after a failed ASAP installation:

1. Open the **deployWbApp** script file in the *ASAP_home***/scripts/** directory.
2. Update the **deployWbApp** script by adding the following option:

   ```
   -Xmx128m
   ```

3. Restart the installer by using the following command:

   ```
   java -Xmx128m -Xms128m weblogic.Deployer -adminurl WL_url -username WL_username -
   name WL_name -targets server_name -source EAR_file_path -upload -deploy
   ```

8

Installing the Order Control Application Client

This chapter describes how to install the Oracle Communications ASAP Order Control Application (OCA) client. Before installing the OCA client, read these chapters:

- ASAP Installation Overview
- Planning Your ASAP Installation
- ASAP System Requirements
- Configuring the Operating System
- Installing and Configuring Oracle Database and Client
- Installing and Configuring Oracle WebLogic Server
- Installing ASAP

About Order Control Application Client Session and User Configuration

Any OCA client can have sessions to one or more ASAP instances. There is a one-to-one relationship between a session and an ASAP instance.

If you are planning to operate more than 500 OCA clients, consider increasing the value of OCA_MAX_SESSIONS parameter in the **ASAP.cfg** file. The default number of sessions is 100.

You may also extend OCA client to connect to multiple ASAP instances by adding new server sessions to OCA client.

You can configure additional OCA client users in the Oracle WebLogic Server. For more information on configuring OCA client users, see the discussion on GUI interface security in *ASAP System Administrator's Guide*.

> **Note:**
>
> If you have errors with the OCA client and security exceptions are generated by the WebLogic Server you may need to update an OCA-related security setting within a server domain.
>
> - In the WebLogic Server Administration Console, navigate to the Security section, and under Realms, click on your realm (For example, myrealm).
> - On the **General** tab, locate the **Check Roles and Policies for entry** and change it to **Web Applications and EJBs protected in DD**.
> - Click **Apply**.
> - Restart your Oracle WebLogic Server.

Types of Installation

You can install the OCA client with **OCA Thin client Web Installer**, using Internet Explorer or Mozilla Firefox (see "Configuring Order Control Application Thin Client (Java Applet)").

Configuring Order Control Application Thin Client (Java Applet)

To set up the OCA think client application, ensure that your browser has the required Java plug-in for the OCA client.

> **Note:**
>
> There can be large performance differences when performing queries which return large numbers of work orders. See *ASAP Compatibility Matrix* for your release to confirm the recommended JRE browser plug-in version.

Adding Order Control Application Thin Client Server Sessions

The following procedure describes how to add new server sessions for a given OCA client.

OCA users that have access privileges to multiple OCA sessions can select one or more sessions to use when performing order queries or audit log queries.

You need to perform the following procedure only if you intend to add new session configurations; ASAP installation automatically creates one session to the OCA JSRP.

To add new server session configurations to the OCA web client application:

1. To open the web-based client, open a web browser and type:

   `http://WebLogic_host:WebLogic_port/ASAP_envID/OCA`

   For example: http://10.15.15.15:7001/AS47/OCA

   WebLogic_host is the WebLogic host on which ASAP is deployed and *WebLogic_port* is the WebLogic port.

   ASAP_envID is the ASAP environment ID. An ASAP system can have multiple ASAP instances with multiple environments.

   The OCA Web Client Configuration screen appears.

   > **Note:**
   >
   > For information on how to open OCA web client using https connection, see *ASAP OCA Client SSL Configuration* (Doc ID 1514874.1) knowledge article on My Oracle Support.

2. Click **Add Server Session Configuration**.
3. Supply the following information.

- **ASAP Environment ID**: The ASAP environment ID is a unique identifier for each ASAP environment within an ASAP instance. Maximum of four alphanumeric characters.
- **Control Database Host**: This is the host name or IP address of the ASAP Database Server.
- **Control Database Port**: Enter Control Database Port.
- **Control Database Name**: The name of the control database for the ASAP instance.
- **Control User Name**: The control database user ID.
- **Control User Password**: The control database user password.
- **WebLogic Administrator**: WebLogic Server system administration login ID.
- **WebLogic Password**: WebLogic Server system administration login password.
- **WebLogic Target Server**: The name of the WebLogic Server for the ASAP instance.

4. Click **Add Server Session**.

   A message appears confirming the successful addition of the server session, or any errors related to the addition of the server session.

5. Click **Continue** to return to the OCA Web Client Configuration screen. The new server session appears in the **Registered Server Sessions** list.

Launching an Order Control Application Thin Client Server Session

To launch an OCA thin client server session:

1. Open a web browser and type:

   `http://WebLogic_Host:WebLogic_port/ASAP_envID/OCA`

   For example: http://10.15.15.15:7001/AS47/OCA

   WebLogic_host is the WebLogic host on which ASAP is deployed and *WebLogic_port* is the WebLogic port.

   ASAP_envID is the ASAP environment ID. An ASAP system can have multiple ASAP instances with multiple environments.

   The OCA Web Client Configuration screen appears.

2. Select a server to launch from the **Select from the following Registered Server Sessions** list.

3. Set the following global parameters:

   - **Language Code**: Language code for the language in which translatable information is returned to and displayed by the client. If applied, all translatable fields are shown in the OCA client native language, as specified by the LANGUAGE variable.
   - **Idle Timeout**: Maximum idle time, in minutes, after which the OCA SRP server terminates the OCA client connection. Set to zero (0) to disable this feature.
   - **Events Supported**: Enables the propagation of SARM events.
   - **Max Open Windows**: The number of detail windows a user can have open at one time.
   - **Default Template**: The template (user or global) that will be selected by default in the Order Entry Wizard after OCA client is launched.

- **Date Format**: The date format that will be displayed in the OCA client including in **work order headers**, the **Work Order Query** window and its results, the **Work Order Release** dialog box, **audit logs**, report generation dialog boxes, and in standard ASAP reports. This date format is also displayed in **OCA Console log** (messages in **OCA Console window**), **New Work Order** window, **Work Order Details** window, and **Events Message Board** window.

4. Click **Launch Selected Server Sessions**.

> **Note:**
>
> The ASAP installer creates a user named **ASAP_admin** that you can use to access the OCA client. Users with the access privileges can change the default user name and password using the WebLogic Server Administration Console.

Configuring the OCA Client on Systems with Multiple Interfaces or IP Addresses

If the Windows computer hosting your OCA client is configured with more than one network interface or IP address or if the hostname of your computer maps to a network address that the WebLogic Server cannot contact (for example when the OCA user is accessing ASAP over a VPN) configure your client as follows.

Configuring the Order Control Application Thin Client (Java Applet)

1. On your Windows system, click **Start**, point to **Settings**, and click **Control Panel**. The Control Panel dialog appears.
2. Double-click the **Java** icon to open the Java Control Panel.
3. In the Java Control Panel, open the **Java** tab.
4. In the **Java Applet Runtime Settings** box, click **View**.

   The **Java Runtime Settings** dialog appears.
5. In the **Java Runtime Versions** table, add the following entry to the **Java Runtime Parameters** column for your preferred version of JRE:

```
-Dweblogic.corba.client.bidir=true
```

> **WARNING:**
>
> Do not overwrite the existing entries.

Localizing the Order Control Application Client

Perform the following procedures to translate OCA client interface components as well as the OCA applet to a language other than English. This procedure does not extend to the HTML page within which the OCA applet runs.

All elements that can be translated are contained in the **OCAResources.properties** file. Specifically, elements that can be translated are:

- Screen titles
- Field labels
- Informational labels (such as the number of work orders found)
- Menus and their contents
- List headers
- OCA-specific error messages
- Tooltips
- OCA-populated drop-down lists

Elements that cannot be translated are:

- Error messages that are generated by the OCA SRP
- Data that appears within fields and data lists
- Data that appears in some drop-down lists populated by systems external to the OCA client

It is not necessary to modify the **OCA.cfg** file.

Configuring Your Locale

To configure your locale:

1. On your Windows system, click **Start**, point to **Settings**, and click **Control Panel**. The Control Panel dialog appears.
2. Double-click the **Regional and Language Options** icon. The Regional and Language Options dialog appears.
3. Select the target locale, such as "French (Canada)".

Configuring Your Browser if Running the OCA Applet

To configure your browser if you are running the OCA applet:

1. Open the web browser and select **Tools** > **Internet Options**.
2. In the General tab, click **Languages**.
3. In the Language Preference dialog, click **Add**.
4. Ensure that "French(Canada)[fr-ca]" and "English(United States)[en-us]" are included in the list.
5. Click **OK**.

Translating the Template File

To translate the template files:

1. Open the **OCAResources.properties** file and save it with a different name. For example, if you are producing a French properties file, you could name it **OCAResources_fr_ca.properties**.

2. Open the **OCAResources_fr_ca.properties** file and translate the values to the right of the equals sign as required. For example:

```
# OCA Frame Temp Action
TMP_ACTION=Ajouter
TMP_ASDLDELAYFAIL=ASDL insuccès retardée
TMP_ASDLRETRY=ASDL réessayage
...
```

3. Package the translated **OCAResources_fr_ca.properties** file into a new JAR file (**ocac_lang.jar**).

   For example, type:

```
jar uvf ocac_lang.jar
```

   For the OCA client (Windows platform), **ocac_lang.jar** is located in **$OCA_ClientHOME\lib**.

   For example, C:\Program Files\Oracle Communications\ASAP\OCAClient\lib.

   If you are running OCA as a web application, you must copy the **OCAResources_fr_ca.properties** to the UNIX platform using FTP, compile it, and package it to **ocac_lang.jar** on the server side. On the server side, **ocac_lang.jar** is normally bundled in **asap*ENV*.ear**.

4. In the Target command, add the "-Duser.language=fr_ca" flag before the classpath flag.

5. Append the **ocac_lang.jar** file to the classpath.

9
ASAP Post-Installation Tasks

This chapter provides instructions for performing Oracle Communications ASAP post-installation tasks.

Overview of ASAP Post-Installation Tasks

Post-installation tasks for ASAP include:

- Configuring ASAP to Support TAF for Oracle RAC
- Installing and Testing a Cartridge
- Using ASAP with Oracle Database 19*c* and NLS_CHARACTERSET WE8ISO8859P15
- Creating a Backup of ASAP, WebLogic Domain, and ASAP Schemas

Configuring ASAP to Support TAF for Oracle RAC

To configure the *ASAP_home***/config/ASAP.cfg** file to take advantage of the Oracle Real Application Cluster (RAC) system functionality and transparent application failover (TAF), use the following procedure:

1. Using a text editor, open the **ASAP.cfg** file in the *ASAP_home***/config/** directory.
2. Configure the entries in **ASAP.cfg** listed in Table 9-1.

Table 9-1 ASAP.cfg Entries

Entry	Description
DB_RETRY_NUMBER	Specify the number of times you want ASAP to retry the connection when the connection to the database fails. For example, `DB_RETRY_NUMBER = 11`.
DB_RETRY_INTERVAL	Specify the interval (in seconds) between attempts for establishing the connection. For example, `DB_RETRY_INTERVAL = 10`.

The following are reliability related configuration parameters in the **ASAP.cfg** file:

```
# ASAP Availability Enhancement Parameters
DB_RETRY_NUMBER = 5 # Manual Retry count on loss of DB Connection
DB_RETRY_INTERVAL = 1 # Sleep period between each manual retry on loss of DB connection
USE_OCI_EVENTS = 1 # Activates OCI_EVENTS on OCI Environment
```

For information about troubleshooting ASAP, see the *ASAP System Administrator's Guide*.

Installing and Testing a Cartridge

This section describes the following procedures related to installing and testing a cartridge:

- Downloading a Cartridge

- Installing a Cartridge

> **Note:**
>
> Sample cartridges are not deployed by default with 7.4.1 installer. You can deploy the sample cartridge **NORTEL_DMS_POTS.sar** manually as a post-installation task.

Downloading a Cartridge

To download a cartridge:

1. Go to My Oracle support website:

   https://support.oracle.com/CSP/ui/flash.html

2. On your workstation, create a repository directory with a name of your choice.
3. Download the cartridge to a temporary directory.
4. Extract the TAR file.

   ```
   tar xvf TAR_file.tar
   ```

   The directory structure should look like the following:

   ```
   repository_directory
     /README.txt
     /installCartridge
     /uninstallCartridge
     /SAR_file.sar
   ```

   This is the minimum required structure. You can enhance this directory structure with additional directories.

Installing a Cartridge

There are two ways of installing a cartridge:

- Installing a Cartridge Using Scripts
- Installing a Cartridge Using Design Studio

Before installing a cartridge, ensure that ASAP is running.

To start ASAP:

1. Start ASAP:

   ```
   start_asap_sys -d
   ```

2. Verify that the Oracle WebLogic Server instance for this ASAP environment is running.

The *ASAP System Administrator's Guide* contains more information on starting the ASAP server, the ASAP Daemon, and the WebLogic Server.

Installing a Cartridge Using Scripts

The **installCartridge** script runs the following tasks:

- Configures the cartridge-specific NE using the Service Activation Configuration Tool (SACT).

- Deploys the cartridge service model (only if the cartridge service model is not yet deployed) using the Service Activation Deployment Tool (SADT).
- Copies the cartridge-specific JAR files to the ASAP environment.
- Using the **PostDeploySarFile** script:
  - Searches for and invokes the **SarPatch** script inside a SAR file.
  - Searches for and populates all SQL files whose name has no **undeploy_** prefix.
  - Searches for all **.tst** work order files and load the work order defined in them.

For information about the SACT refer to the *ASAP Server Configuration Guide*. For more information about SADT and the PostDeploySarFile, refer to the *ASAP Cartridge Development Guide*.

To install the cartridge:

1. Run the *ASAP_home*/**samples/DIT/scripts/installCartridge** script.

   ```
   installCartridge SAR_file.sar
   ```

   Where *SAR_file* is the name of your SAR file.
2. The script prompts you for the values of the following WebLogic Server login parameters:
   - WebLogic Hostname
   - WebLogic HTTP Port
   - WebLogic Login User ID
   - WebLogic Login Password

   The script loads the NEP-NE configuration and the CSDL-ASDL configuration to the SARM database, and loads sample work orders to the database. The script also copies the cartridge-specific jar files to the ASAP environment.
3. Add **${ASAP_BASE}/lib/studio_2_6_0.jar** to the CLASSPATH in the **JInterpreter** file under **$ASAP_BASE/programs** directory.
4. Restart ASAP to upload the cartridge configuration into ASAP.

Installing a Cartridge Using Design Studio

Ensure ASAP and WebLogic Server are started and running. For more information on using Oracle Communications Service Catalog and Design - Design Studio with ASAP, see *Design Studio Help for ASAP*.

To install a cartridge using Design Studio:

1. From the **File** menu, select **Import**, then select **Activation Archive (SAR)** under **Studio Wizards** to import the SAR file. Browse for the path of the SAR file and click **Finish**.
2. Create a new **Service Activation Project**.
3. Define a new **NE Entity** based on the **NE Template** contained in the cartridge provided by Oracle.
4. Ensure that the primary pool of the newly created NE is different from the NE template primary pool. You can modify it, if necessary.
5. Ensure that the test work order provided with the cartridge targets the newly defined NE. If not, then modify the test work orders file(s).

6. Create a new **Activation Environment Project** from the **Studio** menu. See Design Studio documentation for more information.
7. Create **Activation Environment** inside the **Activation Environment Project** and configure the **Connection Details** tab with your Environment ID, Activation version, and WebLogic Server data.
8. Connect to your environment using the **Connect** button.
9. Select the **Cartridge** tab of the **Activation Environment** and click **Add** to add your projects to the environment. The cartridge and the newly created **Service Activation** should appear in the **Cartridges** list.
10. Deploy the **NetworkActivation** (NA) cartridge provided by Oracle. (No NE information is to be deployed with this cartridge, therefore it is not necessary to deploy the **NEP map** info).
11. Deploy the **Service Activation** (SA) project as follows:
    - On the **Cartridge** tab, select the necessary SA cartridge and click **Deploy**.
    - Select the **NEP Map** tab of the Activation Environment. Select the required **NEP** server from **Network Element Processors**. See Studio documentation for more information.
    - Select the SA cartridge from the **Network Element Processor Map** and click **Deploy**.
12. Verify the SADT console to confirm the installation.
13. Go to ASAP environment.
14. Add **${ASAP_BASE}/lib/studio_2_6_0.jar** to the CLASSPATH in the **JInterpreter** file under **$ASAP_BASE/programs** directory.
15. Restart ASAP in order to start working with the cartridge.

Using ASAP with Oracle Database 19*c* and NLS_CHARACTERSET WE8ISO8859P15

Perform the following tasks if you are using Oracle Database 19*c* with **NLS_CHARACTERSET WE8ISO8859P15**.

To enable ASAP to function with the **NLS_CHARACTERSET WE8ISO8859P15**

1. After you install the ASAP server software, you must copy the Oracle 19*c* version of **orai18n.jar** from *Middleware_home***/oracle_common/modules/oracle.nlsrtl** to *ASAP_home***/JLIB**.
2. Add *Middleware_home***/oracle_common/modules/oracle.nlsrtl** to the CLASSPATH for WebLogic Server.
3. Add PRE_CLASSPATH in *domain_home***/bin/setDomainEnv.sh**.
4. Clear the PRE_CLASSPATH in case an application template wants to set it before the larger PRE_CLASSPATH is invoked as shown below:

```
PRE_CLASSPATH=""
export PRE_CLASSPATH

PRE_CLASSPATH="$WL_HOME/oracle_common/modules/oracle.nlsrtl/orai18n.jar"
export PRE_CLASSPATH
```

Creating a Backup of ASAP, WebLogic Domain, and ASAP Schemas

After installing or before upgrading ASAP, create a backup of the ASAP environment, WebLogic Server domain, and ASAP schemas. These backups can be used to recover from system or upgrade failures.

Ensure that you have installed the **zip** and **unzip** utility (or a similar utility such as **tar**) before you start the backup procedures.

The Oracle database server provides the **impdp** and **expdp** utilities to import and export database schemas.

> **Note:**
>
> While backing up ASAP, you must shutdown ASAP and the ASAP WebLogic Server domain. Do not start these backup procedures until the status for all work orders indicate **Completed**, **Cancelled**, or **Failed**, and no work order status indicates **in Loading**, **Initial**, or **in progress**. The Oracle database server must be running.
>
> To minimize service outage while you upgrade, perform the backup during off-peak hours.

Creating a Backup of the ASAP Environment

To create a backup of the ASAP environment, use the following procedure:

1. Source the *ASAP_home*/**Environment_Profile**.

   ```
   . ./Environment_Profile
   ```

2. Shut down ASAP.

   ```
   stop_asap_sys -d
   ```

3. From the directory that contains *ASAP_home*, use the zip utility to generate an ASAP backup file:

   ```
   zip -r ASAP741_BACKUP.zip ASAP_home
   ```

4. Store the backup file in a safe location.

Creating a Backup of the WebLogic Server

To create a backup of the WebLogic Server:

1. Shut down the WebLogic Server using the *domain_home*/**bin/stopWeblogic.sh** script or from the WebLogic Server Administration Console.

   ```
   ./stopWeblogic.sh
   ```

2. From the directory that contains the *domain_home* folder, use the zip utility to generate an WebLogic Server Domain backup file.

   ```
   zip -r WLDOMAIN_BACKUP.zip WeblogicDomain
   ```

> **Note:**
>
> It is also good practice to backup the WebLogic Server software located at *Middleware_home*.

3. Store the backup file in a safe location.

Creating a Backup of the ASAP Schemas

To create a backup of the ASAP Database:

1. Ensure that your database is running.
2. Source the ASAP **Environment_Profile** in the *ASAP_home* folder.

   ```
   . ./Environment_Profile
   ```

3. Use the following command to generate a backup file for the Control, Admin, SARM, NEP, and SRP servers, where *password* is the database system admin password and *filename* is the name you want to give your DMP file.

   ```
   expdp system/password SCHEMAS=$CTRL_USER,$SARM_USER,$ADM_USER,$NEP_USER
   DUMPFILE=filename.dmp
   ```

> **Note:**
>
> You can find the location for the DMP file at the end of the output from this command. For example:
>
> ```
> Dump file set for SYSTEM.SYS_EXPORT_SCHEMA_01 is:
>   /path/dpdump/filename.dmp
> ```
>
> Where *path* is the path to the **dpdump** directory.
>
> If the database is co-resident with ASAP, you must log in with a database user account to access the file. If the database is on a different UNIX machine, you must log in to the UNIX machine with a database user account to access the file.

10 Verifying the ASAP Installation

This chapter describes how to verify that Oracle Communications ASAP is installed correctly.

Checking the State of all Installed Components

You can verify that ASAP is installed by checking the state of all installed components.

To check the state of all installed components:

1. Log in to the WebLogic Server Administration Console.
2. In the left panel, in the Domain Structure section, click **Deployments**.

   The **Summary of Deployments** page appears.
3. Ensure that the managed server is running.
4. If ASAP is installed successfully, the following deployments appear in the **Active** state:
   - asap*envid* (where *envid* is the environment ID for the ASAP instance)
   - cartridge_management_ws
   - SecurityServices

Testing the ASAP Server Installation

The ASAP Server installation provides a Demonstration Service Activation Model named **POTS Service Activation Model** cartridge. You can use the sample model to test the ASAP installation.

To test the ASAP Server installation, start ASAP and send sample work orders to the sample activation cartridge.

It is also recommended that you restart the WebLogic Server before testing the ASAP installation.

To test the installation:

1. From *ASAP_home*, source the **Environment_Profile** file:

   ```
   . ./Environment_Profile
   ```
2. (Optional) Set the **TNS_ADMIN** variable to the location of your **tnsname.ora**. This step is only required if the *Oracle_home***/network/admin/tnsnames.ora** file does not contain the ASAP database information.
3. Start ASAP using the following command:

   ```
   start_asap_sys -d
   ```
4. From the UNIX prompt, enter:

   ```
   status
   ```

   A list of the ASAP servers that are running is displayed.

5. Locate the appropriate executable and enter it at the UNIX command prompt.

   Refer to the README file for **Nortel_DMS_POTS.sar** cartridge for instructions on running ASAP.

   The sample model **POTS Service Activation Model** is installed at **$ASAP_BASE/samples/sadt/DemoInstall/Nortel/DMS/POTS/**.

6. Run the following commands to install the sample cartridge:

```
asapd -start -d -host DAEMON_HOST -port DAEMON_PORT
installCartridge $ASAP_BASE/activationModels/Nortel_DMS_POTS.sar
```

7. To verify that work orders are executing, enter **asap_utils**.
8. When prompted, enter the password to the control server.
9. When prompted, enter 1 to view the service requests in the SARM database. The WO_STAT column displays the status of the work orders.

   The expected results when running in loopback mode are listed below. (If not in loopback mode, the results vary depending upon your specific configuration.)

   - Success (104)
   - Failure (253)
   - In Progress (103)

> **Note:**
>
> If ASAP fails to start, check if the sample cartridge is installed properly and check the logs for more information.

Testing a Cartridge Installation

To test a cartridge installation, you need to know about the network element, services, and basic activation configuration. You may need to perform adjustments to provision a service for a specific NE, network, or connectivity configuration.

You can test the cartridge installation using one of the following methods:

- **Loopback Mode:** Does not actually connect to or send commands to the NE.
- **Live Mode:** Connects to and sends commands to a live NE.

Configuring Loopback and Live Mode Parameters

Set the following variables to test the cartridge in loopback or live testing modes.

Loopback Mode:

Set the parameter listed in Table 10-1 to test the cartridge in loopback mode.

Table 10-1 Loopback Mode Parameter Settings

Configuration Variable	Parameter Settings	Location
LOOPBACK_ON	1 (default setting)	*ASAP_home*/**config/ASAP.cfg**

Live Mode:

Set the parameter listed in Table 10-2 to test the cartridge in live mode.

Table 10-2 Live Mode Parameter Settings

Configuration Variable	Parameter Settings	Location
LOOPBACK_ON	0	*ASAP_home*/**config/ASAP.cfg**

Mapping the *NE_Cartridge_name_HOST*.xml file to NEP Servers

To modify the *NE_Cartridge_name_HOST.xml* file:

1. Create a new source directory with a name of your choice.

   ```
   mkdir New_source_directory
   ```

2. Copy *SAR_file.sar* to this new source directory.

   ```
   cp SAR_file.sar ./New_source_directory
   ```

3. Change directory to *New_source_directory*.

   ```
   cd New_source_directory
   ```

4. Un-jar *SAR_file.sar*. This extracts the contents of the sar file.

   ```
   jar xvf SAR_file.sar
   ```

5. Edit *New_source_directory*/*NE_Cartridge_name_HOST.xml* to map the file to the NEP servers.
6. Create a new sar file at the *New_source_directory* level.

   ```
   CreateSar $PWD
   ```

7. Uninstall the cartridge using *SAR_file.sar*.
8. After you uninstall the cartridge, rename the sar file, so you have a backup copy.
9. Copy the new sar file from *New_source_directory*.
10. Reinstall the cartridge.

Testing the Installation

The following procedure describes the steps required to test the cartridge installation in loopback mode. We recommend that you perform the initial cartridge installation test in loopback mode.

To test the cartridge installation in loopback mode:

1. Stop ASAP by entering the following command at the UNIX prompt:

   ```
   stop_asap_sys -d
   ```

2. Ensure loopback mode is on. See "Configuring Loopback and Live Mode Parameters" for a description of how to set the loopback parameter to *On*.
3. Start ASAP by entering:

   ```
   start_asap_sys -d
   ```

4. Send the sample work orders using the JMS client. See "Sticky and Non-Sticky Requests Supported by Order Balancer"*ASAP System Administrator's guide* for more information.

5. Verify the status of the sample work orders by entering:

   ```
   asap_utils -d l
   ```

   All successful work orders return to 104 state.

   To view the sample work orders provided with the cartridge, refer to the cartridge source.

Viewing the Sample Work Orders

You find the sample work orders under the **sample_wo** directory in the SAR file. The following procedure describes how to view the sample work orders.

To view the sample work orders:

1. Create a repository directory, copy the SAR file to the new directory, and un-jar the SAR file, as described in Step 1 through Step 4 in "Mapping the *NE_Cartridge_name_HOST*.xml file to NEP Servers."
2. Locate and view the sample work order files.

11

Upgrading ASAP

This chapter describes how to upgrade your existing Oracle Communications ASAP system to ASAP. It also includes information about new features, any impact that these new features have with respect to the operation of ASAP, and instructions on how to recover from a failed upgrade.

In this chapter, the ASAP release running on your production or development system is called the *old* release. The release you are upgrading to is called the *new* release.

About Upgrading ASAP

Upgrading to ASAP 7.4.1 consists of the following process:

- Planning the upgrade
- Implementing and testing the upgrade on a development test system
- Preparing to upgrade a production system
- Implementing and testing the upgrade on the production system

The upgrade process includes these tasks:

- Updating the Oracle database and client
- Updating the WebLogic Server and domain
- Updating the ASAP software
- Implement new features

For current patch information, see *ASAP Compatibility Matrix*. For current release information, see "ASAP System Requirements ."

Supported Upgrade Paths

You can directly upgrade to ASAP 7.4.1 from the ASAP 7.4 or ASAP 7.3 release.

Planning Your Upgrade

To plan your upgrade, review the following tasks:

- Identifying Your Upgrade Team
- Identifying Who Is Affected by the Upgrade
- Determining the Impact of New Features
- Estimating How Long the Upgrade Will Take
- Maintaining Access to Services While Upgrading
- Creating Test Environments
- Testing Your Upgraded System

Identifying Your Upgrade Team

Your upgrade team should include the following team members:

- A database administrator to manage the database upgrade and tune the database
- A system integrator to handle new and existing ASAP customizations
- A system administrator to manage the WebLogic Server and ASAP software upgrade
- A UNIX administrator to manage accounts, network setup, and IP configurations

Identifying Who Is Affected by the Upgrade

You should identify who might be affected by the upgrade. For example:

- You might need to give your customer service representatives (CSRs) who send work orders to ASAP advanced notice of any system downtime.
- Tell your system administrators in advance about any changes to the system architecture (for example, Oracle database, client, or WebLogic Server upgrades).
- Train CSR, cartridge developers, or system integrators on any new ASAP functionality that the upgrade may provide that would have an impact on their role.
- Notify Oracle so that Oracle can help you anticipate and avoid problems. Technical support might have additional information about upgrading ASAP or information specific to your implementation.

Determining the Impact of New Features

You might need to make changes to your current system to accommodate additional or modified functionality in the new ASAP release. For example, if the release provides new security functionality, additional system configuration steps may be required.

For more information, see "Upgrade Impacts from ASAP 7.3.0.x or ASAP 7.4.0.x to ASAP 7.4.1."

Estimating How Long the Upgrade Will Take

This is an important consideration because work order processing might be suspended as a result of upgrading ASAP.

The best way to determine how long the database upgrade will take is to run the upgrade on a test system that contains a duplicate of the data in your production system (see "Creating Test Environments").

In general, it takes longer to upgrade large databases with large tables.

Maintaining Access to Services While Upgrading

While upgrading ASAP, you must shutdown ASAP and the ASAP WebLogic Server domain. Do not start the upgrade procedure until the status for all work orders indicates **Completed** and no work order status indicates **in Loading**, **Initial**, or **in progress**.

To minimize service outage while you upgrade, perform the upgrade during off-peak hours.

Creating Test Environments

To test the upgrade process, create a test environment. Use this environment to do the following:

- Test the upgrade process and its results.
- Compare the default behavior of the old and new ASAP releases.
- Determine what customizations you made in the old ASAP release. Any customized ASAP clients and servers (SRP or NEP) or application that provides a programmatic integration with ASAP (using of the APIs described in the *Developer's Reference Guide*) should be identified.

For more information on creating a test environment from an existing ASAP environment, see "Creating a Test Environment from an Existing Environment."

Testing Your Upgraded System

When testing your upgraded ASAP system, verify that:

- The database tables are installed.
- The data in database tables is correct.
- All IP addresses and ports are correct.
- The WebLogic Server starts.
- The SARM starts.
- The NEPs start.
- Users and user permissions are correct.
- Cartridges are imported correctly.
- No changes need to be made to existing cartridges.
- No changes need to be made to the SRT.
- Work orders are being processed correctly.
- NEs are properly provisioned.

Creating a Test Environment from an Existing Environment

When you create a new Oracle Communications ASAP environment, dynamic ASAP data is created automatically. However, if you want to create a new environment that is identical to an existing environment, you must copy the data from the existing ASAP static tables to the new environment.

You may want to create additional identical environments for comparison testing.

Copying Static Data

If you want to move your static data from one database to another, you must first save the data in a flat file and then load the data into the new database. If the data used for the static tables is already stored within flat files, then simply load the data into the tables. All the static data tables are listed by component following these procedures.

To save tables to files (Oracle):

The **ora_dump_sql** script copies the table data to a file named *TABLE NAME***.sql**, where *TABLE NAME* is the name of the table in uppercase characters. If there is more than one table, the script creates a separate file for each one.

1. In a UNIX operating system, type the command line:

   ```
   ora_dump_sql [-d|-n|-t] user ID/password [table name]
   ```

   where [-d|-n|-t] are options that apply to reinserting the data as follows:

   - d – Deletes and replaces with new rows
   - t – Truncates table first
   - n – Inserts new rows, no deletions

   user ID and *password* are your user ID and password for the appropriate component

   [*table name*] is the name of the static table. If you do not specify a table, the script takes all tables for the user and copies them to separate files.

To load data into tables:

1. In a UNIX, type the command line:

   ```
   sqlplus user_name/password filename
   ```

   where:

- *user_name* and *password* are your user name and password for the appropriate component
- *filename* is the name of the file where the static data is stored

Control Schema Static Data tables

- **tbl_alarm_center**
- **tbl_appl_proc**
- **tbl_classA_secu**
- **tbl_classB_secu**
- **tbl_code_list**
- **tbl_component**
- **tbl_db_threshold**
- **tbl_event_type**
- **tbl_fs_threshold**
- **tbl_name_value_pair**
- **tbl_system_alarm**
- **tbl_unid**

Service Activation Request Manager Schema Static Provisioning Translation Tables

- **tbl_asdl_config**
- **tbl_asdl_parm**

- **tbl_csdl_asdl**
- **tbl_csdl_config**
- **tbl_nep_asdl_prog**
- **tbl_nep_jprogram**
- **tbl_nep_program**
- **tbl_program_source**

Service Activation Request Manager Schema Static NE Configuration Tables

- **tbl_clii_route**
- **tbl_comm_param**
- **tbl_cp_mux**
- **tbl_err_threshold**
- **tbl_host_clii**
- **tbl_ne_config**
- **tbl_ne_event**
- **tbl_ne_monitor**
- **tbl_nep**
- **tbl_nep_mux**
- **tbl_nep_rte_asdl_nxx**
- **tbl_q3_agent_name**
- **tbl_resource_pool**
- **tbl_user_err_threshold**

Upgrade Impacts from ASAP 7.3.0.x or ASAP 7.4.0.x to ASAP 7.4.1

This section explains any important system changes introduced by an upgrade.

New features and new functionality are described in the ASAP Release Notes.

Support for Oracle Fusion Middleware 14*c*

ASAP 7.4.1 supports Oracle WebLogic Server 14*c*. To upgrade your WebLogic Server, you must also upgrade your JDK to the version specified in *ASAP Compatibility Matrix*.

Support Oracle Database Server 19*c*

ASAP 7.4.1 supports Oracle Database Server 19*c*. You need to upgrade Oracle Database Server to the version specified in *ASAP Compatibility Matrix*.

Support for Oracle Database Client 19*c*

ASAP 7.4.1 supports Oracle Database Client 19*c*. You need to install only one version of the client. See *ASAP Compatibility Matrix* for more information.

Schema Changes

There are no schema changes for various possible upgrades between ASAP 7.3 and ASAP 7.4.

Upgrading from ASAP 7.3.0.x or ASAP 7.4.0.x to ASAP 7.4.1

Table 11-1 describes the upgrade methods for ASAP software component upgrade methods.

Table 11-1 ASAP Software Component Upgrade Methods

Software	Upgrade Method	Description
Oracle database	In-place or out-of-place (during the database installation, using **dbua** after installing the database, or using the **expdp** and **impdp** utilities)	You can either upgrade your existing ASAP database when you install the new version of the database software, or run *Oracle_home***/bin/dbua** after you have installed the new database software. You can also create a new database and use the **expdp** or **impdp** utilities to import the old database into the new one. You may perform an in-place upgrade (into the same *Oracle_home*) or an out-of-place upgrade (into a separate *Oracle_home*). You may also upgrade your database from a single instance Oracle Database to an Oracle RAC database. Refer to the *Oracle Database Upgrade Guide* for recommendations and procedures. In either case, Oracle recommends that you back up your database before the upgrade. For database backup and restore procedure, please refer to "Creating a Backup of the ASAP Schemas" and "Rolling Back the ASAP Database."
Oracle client	New installation	Install a new version of the Oracle client when you upgrade your database. Download and install the software as described in "Downloading and Installing the Oracle Client Software."
Oracle WebLogic Server	In-place (using **reconfig.sh** script)	Locate the Oracle WebLogic Server software distribution file on the Oracle software delivery website. The WebLogic Server distribution file is located with the ASAP software components. Also download and install the version of Java for the operating system you are using for your WebLogic Server software. See *ASAP Compatibility Matrix* for Java version requirements for your platform. Run *Middleware_home***/oracle_common/common/bin/ reconfig.sh** to activate the **WebLogic Server Reconfiguration Wizard**.
ASAP software	In-place (using the ASAP installer file from the old installation)	Run the ASAP installer after you have upgraded all other ASAP software components. Run the RPM installer to upgrade ASAP files. Then run the upgradeASAP script to configure ASAP to latest verison. Download the ASAP installation files as described in "Downloading the ASAP Installation Files."

> **Caution:**
>
> Before you start the upgrade procedure, you must backup your WebLogic Server domain, ASAP, and Oracle database (see "Creating a Backup of ASAP, WebLogic Domain, and ASAP Schemas"). The upgrade process for these software components is in-place, meaning the upgrades are applied to the existing software. If the upgrade fails, you must use these backups to roll back your upgrades (see "About Rolling Back ASAP").

To upgrade from ASAP 7.3.0.x or 7.4.0.x to ASAP 7.4.1:

1. Stop all the ASAP servers except WebLogic server.
2. Backup ASAP Directory.
3. Download the ASAP 7.4.1 RPM Installer. For more information, see Downloading the ASAP Installation Files
4. Exit the current ASAP kernel.
5. Open new kernel and run the following command to upgrade the installed RPM package:

   ```
   sudo rpm -U --prefix ASAP_home asap-installer-7.4.1.0.0-Bxxx.x86_64.rpm
   ```

   ASAP_home is the existing ASAP installation directory path where the current version is installed.
6. If you are upgrading from ASAP 7.3.0, do the following:
   a. Go to the ASAP directory.
   b. Source Environment_Profile.
   c. Export *TNS_ADMIN*.
   d. Go to the Scripts directory.
   e. Run the following script:

      ```
      asap741.wl.resource.update.sh
      ```

   f. Exit from the kernel.
   g. Start a new kernel and restart WebLogic server.
7. To set the environment variables, run the following commands:

   ```
   export JAVA_HOME_ASAP=Java_17_path
   export TNS_ADMIN=TNS_path
   ```

8. Rename and update the **sampleUpgradeConfiguration.properties** file located at *ASAP_home***/config** to provide the following details:

Table 11-2 sampleUpgradeConfiguration.properties File

Name	Description	Required
ORACLE_HOME	Specifies the name of the base directory of the Oracle database client program. The ASAP installation generates an ORACLE_HOME UNIX environment variable based on the directory name. The default value is Null.	Yes
ANT_HOME	Specifies the path to ant program.	Yes
WL_HOME	Specifies the path to the WebLogic installation directory.	Yes
WLS_ADMIN_PORT	Specifies NON-SSL Port if WLS_SSL_ENABLED is false. Specifies SSL Port if WLS_SSL_ENABLED is true.	Yes
WLS_SERVER	Specifies the Oracle WebLogic server. The default value is **AdminServer**.	Yes
WLS_SSL_ENABLED	Set to **true** if SSL is enable. The default value is **false**.	No
KEYSTORE_FILE	Specifies the path of the SSL KeyStore file.	Yes, if SSL is enabled
GRPC_SSL_ENABLED	Specifies if SSL is enabled for GRPC. The valid values are: • 1 - Enabled • 0 - Disabled	No
GRPC_AUTH_ENABLED	Specifies if the authentication is enabled for GRPC. The valid values are: • 1 - Enabled • 0 - Disabled	No
SSL_IDENTITY_PKCS12_LOCATION	Specifies the absolute path of SSL identity certificate in the pkcs12 format.	Yes if GRPC_SSL_ENABLED is enabled
SSL_TRUST_PKCS12_LOCATION	Specifies the absolute path of SSL trust certificate in the pkcs12 format.	Yes if GRPC_SSL_ENABLED is enabled

9. Run the upgrade script using the following commands:

```
cd scripts
upgradeASAP -properties ../config/sampleUpgradeConfiguration.properties
cd..
source Environment_profile
```

10. Uninstall the previous versions of **NORTEL_DMS_POTS.sar** (if present in the earlier versions of ASAP) and reinstall the **NORTEL_DMS_POTS.sar** cartridge as the state table based cartridge is changed to Java cartridge in ASAP 7.4.1. Use the following commands to uninstall the state table based cartridge and reinstall the Java based cartridge **NORTEL_DMS_POTS.sar**:

```
asapd -start -d -host <DAEMON_HOST> -port <DAEMON_PORT>
uninstallCartridge $ASAP_BASE/activationModels/Nortel_DMS_POTS.sar
installCartridge $ASAP_BASE/activationModels/Nortel_DMS_POTS.sar
asapd -stop -d -url <DAEMON_HOST>:<DAEMON_PORT>
```

> **Note:**
>
> ASAP 7.4.1 does not support state table based cartridges and these cartridges must to be uninstalled if present after you upgrade to ASAP 7.4.1.

11. Start ASAP servers.

Migrating Order Data from ASAP 7.3.0.x or 7.4.0.x to ASAP 7.4.1

To migrate order data from ASAP 7.4.0.x to ASAP 7.4.1:

1. Install ASAP 7.4.1 For more information, see "Installing ASAP".

> **Note:**
>
> ASAP installation should have the same ENV_ID, database users credentials, default users credentials, port numbers, and cartridges deployed in the 7.3 deployment.
>
> The values of ENV_ID and port numbers are present in the ASAP installation directory.

2. Use the **expdp** utility to export ASAP 7.3 database (12*c*) schemas. For more information, see "Creating a Backup of the ASAP Schemas". Oracle database server provides the **impdp** and **expdp** utilities to import and export database schemas.
3. Use the following commands to drop the ASAP users from 19*c* database:

```
drop user admenvid cascade;
drop user sarmenvid cascade;
drop user ctrlenvid cascade;
drop user nepenvid cascade;
drop user srpenvid cascade;
```

   Where *envid* is the environment ID for your ASAP instance.

   For more information see "Rolling Back the ASAP Database".
4. Use the **impdp** utility to import 12*c* database schemas into 19*c* database. For more information see "Rolling Back the ASAP Database".
5. Update the following tables:
   a. Update hostname in `tbl_listeners` of the CTRL database by running the following command:

```
update tbl_listeners set HOST_NAME='asaphost';
```

      where *asaphost* is the name of the host in which ASAP is deployed.
   b. Update the hostname in the `TBL_ASAP_SRP` table of the SARM by running the following command:

```
update TBL_ASAP_SRP set HOST_NAME='asaphost';
```

c. Update the **SRP_HOST_NAME** field in the **TBL_ASAP_SRP** table by running the following command:

```
update TBL_ASAP_SRP set SRP_HOST_NAME='asaphost';
```

6. Restart ASAP.

Upgrading ASAP from a Single Database to an Oracle RAC Database

The following sections describe additional upgrade procedures required if you are upgrading from a single Oracle database instance to an Oracle RAC database instance.

Upgrading the Oracle Database, WebLogic Server, and ASAP Servers

To upgrade the Oracle Database and ASAP Servers:

1. Upgrade the single instance database (see Table 11-1).
2. Upgrade ASAP as described in "Upgrading from ASAP 7.3.0.x or ASAP 7.4.0.x to ASAP 7.4.1."
3. Stop ASAP.
4. Migrate the upgraded single instance database to an Oracle RAC database. For instructions, see the Oracle Database documentation.

Creating WebLogic Generic and Multi Data Sources to the Oracle RAC Database

After upgrading ASAP and migrating the Oracle Database content to an Oracle RAC database, you must delete the original WebLogic data sources that connected to the single database instance and create new WebLogic data sources as listed in Table 11-3. Each Oracle RAC database requires a single generic data source that are both members of a common multi data source.

Table 11-3 ASAP Data Source Migration

Original Data Source Name and JNDI Name	New Data Source Name and JNDI Name
envid.ADMDataSource System.*envid*.ApplicationType.ServiceActivation.Application.1-0;7-3;ASAP.Comp.ADMDataSource	Generic data sources: • *envid*.ADMDataSource-1 System.*envid*.ApplicationType.ServiceActivation.Application.1-0;7-4;ASAP.Comp.ADMDataSource-1 • *envid*.ADMDataSource-2 System.*envid*.ApplicationType.ServiceActivation.Application.1-0;7-4;ASAP.Comp.ADMDataSource-2 Multi Data Source: • *envid*.ADMDataSourcenew • System.*envid*.ApplicationType.ServiceActivation.Application.1-0;7-4;ASAP.Comp.ADMDataSource

Table 11-3 (Cont.) ASAP Data Source Migration

Original Data Source Name and JNDI Name	New Data Source Name and JNDI Name
envid.CTRLDataSource System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-3;ASAP.Comp.CTRLDataSource	Generic data sources: • *envid*.CTRLDataSource-1 • System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.CTRLDataSource-2 Multi Data Source: • *envid*.CTRLDataSource • System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.CTRLDataSource
envid.NEPDataSource System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-3;ASAP.Comp.NEPDataSource	Generic data sources: • *envid*.NEPDataSource-1 System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.NEPDataSource-1 • *envid*.NEPDataSource-2 System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.NEPDataSource-2 Multi Data Source: • *envid*.NEPDataSource • System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.NEPDataSource
envid.SARMDataSource System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-3;ASAP.Comp.SARMDataSource	Generic data sources: • *envid*.SARMDataSource-1 System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.SARMDataSource-1 • *envid*.SARMDataSource-2 System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.SARMDataSource-2 Multi data source: • *envid*.SARMDataSource • System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.SARMDataSource
envid.SRPDataSource System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-3;ASAP.Comp.SRPDataSource	Generic data sources: • *envid*.SRPDataSource-1 System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.SRPDataSource-1 • *envid*.SRPDataSource-2 System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.SRPDataSource-2 Multi data source: • *envid*.SRPDataSource • System.*envid*.ApplicationType.ServiceActivation. Application.1-0;7-4;ASAP.Comp.SRPDataSource

Table 11-4 Additional Settings for Generic Data Sources

Field	Tab	ADM Data Source Value	SARM/NEP/SRP/CTRL Data Source Value
Row Prefetch Enabled	General	Enabled	Enabled
Row Prefetch Size	General	200	200

Table 11-4 (Cont.) Additional Settings for Generic Data Sources

Field	Tab	ADM Data Source Value	SARM/NEP/SRP/CTRL Data Source Value
Driver Class Name	Connection Pool	oracle.jdbc.driver.OracleDriver	oracle.jdbc.driver.OracleDriver
Initial Capacity	Connection Pool	1	3
Maximum Capacity	Connection Pool	5	25
Capacity Increment	Connection Pool	1	3
Test Connections On Reserve	Connection Pool	Enabled	Enabled
Test Frequency	Connection Pool	0	0
Shrink Frequency	Connection Pool	0	0
Remove Infected Connections Enabled	Connection Pool	Disabled	Disabled

To create new generic and multi data sources to an Oracle RAC database:

1. Log in to the WebLogic Server.

   The WebLogic Server Administration Console appears.
2. Click **Deployments**.
3. Select all **Deployments**
4. Click **Stop**.
5. Click **Force Stop Now**.
6. Expand **Services**.
7. Click **Database Sources**.
8. Select all the original data sources listed in Table 11-3 except for the Admin data source. Do not select the RPC data source.
9. Click **Delete**.
10. You must create a generic data source for each Oracle RAC database. Click **New**.
11. Click **Generic Data Source**.

    The Create a New JDBC Data Source screen appears.
12. In the **Name** field, add the first new data source name listed in Table 11-3. For example, *envid*.ADMDataSource-1.
13. In the **JNDI Name** field, add the first new JNDI name listed in Table 11-3. For example, System.*envid*.ApplicationType.ServiceActivation.Application.1-0;7-3;ASAP.Comp.ADMDataSource-1.
14. From the **Database Type** list, select **Oracle**.
15. Click **Next**.
16. From **Database Drive** list, select **Oracle Driver for RAC Service-Instance connections; Version 10: and later**.
17. Click **Next**.
18. Click **Next**.

19. Do the following:
    a. In the **Service Name** field, enter the service name of the Oracle RAC database you want this data source to connect to.
    b. In the **Database Name** field, enter the database name for the Oracle RAC database you want this data source to connect to.
    c. In the **Hostname** field, enter the host name for the Oracle RAC database you want this data source to connect to.
    d. In the **Port** field, enter the port number for the Oracle RAC database you want this data source to connect to.
    e. Enter the **Database User Name** for the Schema you want to create a data source for.
    f. Enter the **Password** for the ASAP schema you want to create a data source for.
    g. Confirm the **Password** for the ASAP schema you want to create a data source for.
20. Click **Next**.
21. Edit the URL field as follows:

    ```
    jdbc:oracle:thin:@//ip_address:port/service_name
    ```

    where *ip_address*, *port*, and service_name are the IP address, port number, and service name for an Oracle RAC database.
22. Click **Test Configuration**.
23. Do the following:
    - If the test succeeds, click **Next**.
    - If the test did not succeed, review the steps in this procedure and check for configuration errors.
24. In the Select **Targets** section, select **Admin** if ASAP is installed in a single Admin server or **Managed Server** if ASAP is installed with an Admin server and a managed server.
25. Click **Finish**.
26. Select the created data source.
27. On the **Transaction** tab, deselect the **Supports Global Transactions** check box.
28. Click **Save**.
29. Repeat steps 6 to 28 for the second new data source name listed in Table 11-3. For example, *envid*.ADMDataSource-2 System.*envid*.ApplicationType.ServiceActivation.Application.1-0;7-3;ASAP.Comp.ADMDataSource-2.
30. You must create a multi data source for Oracle RAC that references these two data sources. Click **New**.
31. Click **Multi Data Source**.
32. In the **Name** field, enter the multi data source name listed in Table 11-3.
33. In the **JNDI Name** field, enter the multi data source JNDI name listed in Table 11-3.
34. From the **Algorithm Type** list, select **Failover**.
35. Click **Next**.
36. In the Select **Targets** section, select **Admin** if ASAP is installed in a single Admin server or **Managed Server** if ASAP is installed with an Admin server and a managed server. The target server should be the same one used for the original data sources.

37. Click **Next**.
38. Click **Next**.
39. From the **Data Source Available** section, select the two generic data sources that correspond to the multi data source node you are creating.
40. Click the right arrow to choose them.
41. Click **Finish**.
42. Repeat steps 6 to 41 for all remaining generic and multi data sources listed in Table 11-3.
43. Update the newly created generic data source values using the values shown in Table 11-4.
44. Click **Services**.
45. Click **Persistent Stores**.
46. Click *envid***.JMSJDBCStore**.
47. In the **Target** list, do one of the following:
    - Select **AdminServer** if ASAP is installed on a single administration server.
    - Select **ManagedServer** if ASAP is installed on an administration server with one or more managed servers.
48. In the **Data Source** field, select the new multi data source for the Admin server. This step only applies to the Admin server. All other servers used JDBC. The target server should be the same one used for the original data sources.
49. Click Save.
50. Click **Database Sources**.
51. Select the old Admin data source.
52. Click **Delete**.
53. Restart the WebLogic server.
54. Log in to the WebLogic server.

    The WebLogic Server Administration Console appears.
55. Click **Deployments**.
56. Select all **Deployments**
57. Click **Start**.
58. Select **Service All Requests**.

Configuring the ASAP Environment for the Oracle RAC Database

To configure the ASAP environment for the Oracle RAC database:

1. From the ASAP environment, source the *ASAP_home***/Environment_Profile**.
2. Open the **Environment_Profile** with a text editor.
3. Search for **DSQUERY**.
4. Change the value for **DSQUERY** to the new Oracle RAC database service name.
5. Save the **Environment_Profile**.
6. Open *ASAP_home***/ASAP.properties** with a text editor.

7. Change the values for **DB_CONNECT** to the new Oracle RAC database tnsnames.ora DB_CONNECT values.
8. Add the **RACDB_CONNECT_STR** parameter and set it to the Oracle RAC database connection string. This connection string repeats the IP addresses and port numbers for the Oracle RAC databases listed in the **tnsnames.ora** file in a different format. For example:

   ```
   RACDB_CONNECT_STR=host1:port:servicename,host2:port:servicename
   ```

   Where *HOST1* and *HOST2* are the name or IP address of your Oracle RAC databases, *port* is the port number for the Oracle RAC databases, and *servicename* is the service name for the Oracle RAC databases.
9. Restart ASAP.

Upgrading Stored Procedures for ASAP Servers

To upgrade ASAP server stored procedures:

1. Change to the **$ASAP_BASE/isql** directory for the new version of ASAP:

   ```
   cd $ASAP_BASE/isql
   ```
2. Run the following commands to upgrade the stored procedures:

   ```
   sqlplus $SARM_USER/sarm_db_password < sarm_procs.sql
   sqlplus $SARM_USER/sarm_db_password < oca_sarm_procs.sql
   sqlplus $CTRL_USER/control_db_password < control_procs.sql
   sqlplus $ADM_USER/admin_db_password < admin_procs.sql
   sqlplus $ADM_USER/admin_db_password < aims_procs.sql
   sqlplus $ADM_USER/admin_db_password < oca_admin_procs.sql
   sqlplus $SRP_USER/srp_db_password < srp_emul_procs.sql
   sqlplus $NEP_USER/nep_db_password < asc_nep_procs.sql
   ```
3. Check the upgraded database object status as follows:

   ```
   $ASAP_BASE/db_migration/oracle_from_asap_ver_to_asap_versionL
   /scripts/check_db_obj_status.ksh
   ```

   The status of all database objects should be VALID. For example:

   ```
   $ASAP_BASE/db_migration/oracle_from_asap_ver_to_asap_versionL
   /scripts/check_db_obj_status.ksh -U $SRP_USER -P srp_db_password
   ```

About Rolling Back ASAP

You can roll back to a previous version of ASAP by using the backup files that you created after you installed ASAP or before you started an upgrade (see "Creating a Backup of ASAP, WebLogic Domain, and ASAP Schemas").

If you are restoring a backup created on one machine to a different machine, ensure the following:

- The new machine name must be identical to the old machine name.
- The new user accounts and permissions should be identical to the old user accounts and permissions.
- The IP addresses for the new box must be updated in the ASAP environment, WebLogic Server domain, and ASAP database servers.

Ensure that you have installed the *zip* and *unzip* utility before you start the roll back procedures for the ASAP environment and WebLogic Server Domain.

The Oracle Database Server provides the **impdp** and **expdp** utilities to import and export database schemas.

Rolling Back the ASAP Environment

To roll back the ASAP environment, use the following procedure.

1. From the directory that contains the old *ASAP_home* folder, use the **unzip** utility to unzip an ASAP backup file.

   ```
   unzip ASAP741_BACKUP.zip
   ```

Rolling Back the WebLogic Server Domain

To roll back the WebLogic Server domain, use the following procedure:

1. From the directory that contains the old *domain_home* folder, use the **unzip** utility to unzip a WebLogic Server domain backup file.

   ```
   unzip WLDOMAIN_BACKUP.zip
   ```

Rolling Back the ASAP Database

To roll back the ASAP Database, use the following procedure:

1. Source the ASAP **Environment_Profile** in the *ASAP_home* folder (where *ASAP_home* is the location of your ASAP server installation directory).

   ```
   . ./Environment_Profile
   ```

2. Drop all database users, if you are importing your database back into an environment where those users were already created.

   a. Log in to your database using ***/*as sysdba** for the user-name.

   b. Enter the following commands to drop the ASAP users:

      ```
      drop user admenvid cascade;
      drop user sarmenvid cascade;
      drop user ctrlenvid cascade;
      drop user nepenvid cascade;
      drop user srpenvid cascade;
      ```

      Where *envid* is the environment ID for your ASAP instance.

   c. Log out of your database.

      ```
      exit
      ```

3. Use the following command to restore the backup file for the Control, Admin, SARM, NEP, and SRP servers. Where *password* is the database system admin password and *filename* is the name that you gave your DMP file when you created the backup (see "Creating a Backup of the ASAP Schemas").

   ```
   impdp system/password SCHEMAS=$CTRL_USER,$SARM_USER,$ADM_USER,$NEP_USER,$SRP_USER
   DUMPFILE=filename.dmp
   ```

12

Uninstalling ASAP

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

About Uninstalling ASAP

You use the RPM to uninstall ASAP. See "Re-Running the ASAP Server Installation" for uninstalling ASAP and its components through RPM.

Uninstalling ASAP WebLogic Server Components

The ASAP WebLogic Server uninstaller removes all of the components and features of ASAP on the WebLogic Server with the exception of the cartridge management component. This component is shared by other Oracle products and should be deleted manually using the WebLogic Server Administration Console.

The uninstaller also removes all changes or customizations to your ASAP WebLogic Server components.

To remove ASAP UNIX server components, see "Uninstalling ASAP UNIX Servers."

> **Note:**
>
> You must uninstall the ASAP WebLogic Server before you uninstall the ASAP UNIX server components.

To uninstall the ASAP WebLogic Server components:

1. Stop the WebLogic Server.
2. Run the following command on domain to remove the domain directory:

   ```
   rm- rf
   ```

Uninstalling ASAP UNIX Servers

You remove the ASAP WebLogic server components and then uninstall the ASAP UNIX servers. To remove ASAP WebLogic server components, see "Uninstalling ASAP WebLogic Server Components ."

To uninstall the ASAP Server installation:

1. Stop the WebLogic server:

   ```
   cd domain_home/bin
   ./stopWebLogic.sh -U wls_admin_user
   ```

2. Remove the WebLogic server components.
3. Drop the Database as appropriate.

4. Remove the installed RPM package by running the following commands:

```
#Get the name of the rpm
rpm -q asap-installer
#Remove the RPM package
rpm -ev rpm_name
```

5. Remove all the directories and files that were created by ASAP installation by running the following command:

```
rm -rf
```

Uninstalling a Cartridge

There are two ways of uninstalling a cartridge:

- Uninstalling a Cartridge Using Scripts
- Uninstalling a Cartridge Using Design Studio

Uninstalling a Cartridge Using Scripts

Run the uninstallation script **uninstallCartridge** to uninstall the cartridge. The script runs the following tasks:

- Deconfigures cartridge-specific NEs using the SACT.
- Undeploys the cartridge service model (only if the cartridge service model is already deployed) using the SADT.
- Removes the cartridge-specific jar files and **cpp** library file from the ASAP environment.

For more information on the SACT and the SADT, refer to the *ASAP System Administrator's Guide.*

To uninstall the cartridge:

1. Run the **uninstallCartridge** script. At the prompt, type:

```
uninstallCartridge SAR_file.Timestamp.sar
```

2. The script prompts you to enter the NEP server name.
3. The script prompts you for the values of the following parameters:
   - WebLogic Hostname
   - WebLogic HTTP Port
   - WebLogic Login User ID
   - WebLogic Login Password

The script unloads the NEP-NE configuration and CSDL-ASDL configuration from SARM database. It also removes the cartridge specific jar files from the ASAP environment.

Uninstalling a Cartridge Using Design Studio

To uninstall a cartridge using Design Studio:

1. Connect to your environment using the **Connect** button.

2. Select the cartridge from the Environment Cartridge list in Design Studio and click **Undeploy**.
3. Verify the Environment Cartridge list. If the check box for the cartridge is unchecked, it means that the cartridge is uninstalled.

A

Pre-Production Checklist

This appendix contains a checklist of items that should be checked or modified before moving Oracle Communications ASAP to production.

Table A-1 provides a pre-production checklist for WebLogic Server configuration.

Table A-1 Pre-production Checklist – WebLogic Server Configuration

Checklist Item	Description	Reference
Design applications to handle undelivered messages	The Admin database contains the WLStore table, a JDBC table that is populated and managed by WebLogic Server to maintain persistence in their JMS destinations. Oracle recommends that you manually clean these tables during development if you want to start with a fresh environment. Otherwise, old undelivered messages will occupy the queues and may cause problems when trying to debug or test. For production environments, you should design applications to handle undelivered messages. For instance, you can configure an error destination to use if a message fails to be delivered after a configurable number of attempts.	For more information, see the latest Oracle Fusion Middleware WebLogic Server documentation.

In production environments, Oracle recommends that you adjust the **ASAP.cfg** configuration variables listed in Table A-2.

> **Note:**
>
> You should also review "ASAP Configuration Parameters" in the *ASAP System Administrator's Guide* to determine whether configuration parameters other than the ones listed in the table below should be adjusted to suit your ASAP implementation.

Table A-2 Pre-production Checklist – ASAP.cfg

Checklist Item	Description	Reference
Set up database administrative routines. Involves the following parameters: DB_ADMIN_ON DB_ADMIN_TIME DB_ADMIN_PROC DB_ADMIN_PROC_PARAM	Database purging can be performed for the SRP, SARM, and control databases, but is most commonly performed in the SARM as this is where the majority of the work order information is stored. The SRP and SARM databases maintain a history of all work orders received, while the ASAP control database maintains a history of alarms, events, performance, and process information. The purging of the SRP and SARM is usually based on work order age. The purge age is usually determined by the amount of available disk space. Usually, only orders that have been completed for a certain amount of time are purged.	See "Database purging" in the *System Administrator's Guide*. See also "ASAP Configuration Parameters" in the *System Administrator's Guide*.

Table A-2 (Cont.) Pre-production Checklist – ASAP.cfg

Checklist Item	Description	Reference
MAX_ORDERS_IN_PROGRESS	This is the maximum number of orders that the SARM allows to be in progress at any given time. This is to limit the memory requirements of the SARM should there be large numbers of orders in progress for long periods of time. If zero, this check is disabled. Default = 40. In production environments, ensure that the MAX_ORDERS_IN_PROGRESS parameter is not set to 0. A setting of 0 specifies no upper limit on memory consumption and can result in significant performance consequences.	See "ASAP Configuration Parameters" in the *System Administrator's Guide*.
DIAG_LINE_FLUSH	If set, it determines whether the diagnostic file output is flushed to disk at the end of each diagnostic line. If disabled, the diagnostics are only flushed to the diagnostic file if a PROGRAM or SANITY level diagnostic message is written, or if the I/O buffer is flushed by the operating system. You can change the treatment of diagnostic files parameter by setting this configuration parameter or changing the diagnostic line flush flag of the application server by using the diag_line_flush API RPC/Registered Procedure to the particular server. Default value is one. Disable (set to 0) in a production environment as it causes considerable performance overhead in disk activity.	See "ASAP Configuration Parameters" in the *System Administrator's Guide*.
LANGUAGE_DUMP_ON	Determines whether the language buffer being transmitted or received should be logged as a low level diagnostic in the server's diagnostic log file. The primary use is for debugging such language buffers, and therefore, is generally disabled in production environments.	See "ASAP Configuration Parameters" in the *System Administrator's Guide*.
WO_AUDIT_LEVEL	Ensure that the audit level is not set to 4. A level of 4 should not be used in production environments as this level of auditing may result in degraded performance.	See "ASAP Configuration Parameters" in the *System Administrator's Guide*.
MEMORY_LOGGING	Determines whether diagnostic memory management RPCs (mem_usage and mem_stats) are created in the server as well as the degree of memory management diagnostic logging to be performed while the server is running. Ensure that this parameter is set to 0 in production systems, this configuration parameter should not be set as it imposes performance overhead.	See "ASAP Configuration Parameters" in the *System Administrator's Guide*.

Ensure that production environments observe the log and diagnostic setting recommendations listed in Table A-3.

Table A-3 Pre-production Checklist – Log and Diagnostic Settings

Checklist Item	Description	Reference
Location of log directories	Local versus NFS-Mounted File Systems for Diagnostic Files ASAP diagnostic files on NFS-mounted file systems increase network traffic and slow down disk I/O. For production systems, the log directories should be local, not NFS mounted.	See *System Administrator's Guide*.
Set up purging of logs and diagnostics	This requires a cron script that clears alarm entries, event logs and process information.	See "Database purging" in the *System Administrator's Guide*.
Set the diagnostic level	SANITY_LEVEL – Used by the application for high-level diagnostics. This level of diagnostic messages provides user information about the processing of the system. It is used for low level diagnostic messages. A production application has its diagnostic level set at either PROG or SANE in tbl_appl_proc. PROGRAM_LEVEL – This is primarily used to generate error messages when the application is running in a production environment.	See "DIAG_LEVEL abstract data type" in *Developer Reference*. See also "ASAP security features" in the *System Administrator's Guide*.

B

Sample Configuration File

ASAP offers a sample configuration properties file to configure WebLogic, ASAP, and Database. The sample configuration properties file can be found in **$ASAP_HOME/config** directory with the name **sampleASAPConfiguration.properties**. You must update the file with the values for database and WebLogic configurations. Oracle recommends to rename the file so that it can be re-used for upgrade.

> **Note:**
>
> Ensure that there are no trailing and leading whitespaces in the values set for the properties in the **sampleASAPConfiguration.properties** file.

The sample configuration file is as follows:

```
##########################################################################
##  Configure ASAP UNIX Environment Variables
##########################################################################
##  Type the name of the base directory of the Oracle database client
program. The ASAP installation generates an ORACLE_HOME UNIX environment
variable based on the directory name.
ORACLE_HOME=
## Provide the path to ant program
ANT_HOME=
##  Set the ASAP mode that you require. You can run ASAP in the Production
(PROD) or Development (TEST) mode. ASAP loads static provisioning
configuration information from the database based on mode.
ASAP_SYS=TEST
##  Type the ASAP environment ID (maximum 4 alphanumeric characters). The
ASAP environment ID is a unique identifier for each ASAP environment as one
system  can have multiple ASAP instances.
ENV_ID=

##########################################################################
##  Configure ASAP Database
##########################################################################
##  The ASAP installation program retrieves the name of the Oracle RDBMS
Server from tnsname.ora file on the installed Oracle client program.
##  Name of Oracle RDBMS Server :
DSQUERY=
##  Oracle Server DBA User Name :
DB_USER=
##  Oracle Server DBA Password :
DB_PASSWORD=
##                          ASAP Database Users Creation
##                         ------------------------------
##  CREATE_USERS=true:  The database users will be created during the ASAP
configuration.
```

```
##  CREATE_USERS=false: The database users must be created by the user
beforehand.
CREATE_USERS=true
##  Is this an Oracle RAC database :
RACDB=false
##  RAC Database Connection String, Format:
Host1:Port1:ServiceName,Host2:Port2:ServiceName,... :
RACDB_CONNECT_STRING=

#############################################################################
##  Configure ASAP Database Table spaces
#############################################################################
##Specify the required table space for each ASAP server. The recommended
table space set up for the SARM server is SARM_DATA for the data table space
and SARM_INDEX for the index table space.
#The recommended table space for all other ASAP servers is DATA for data
table space and INDX for index table space. ASAP database scripts populate
the database schema based on the defined values.
##  ADM
ADMDB_PARAMETERS=ADM, DATA, indx
##  Control
CTRLDB_PARAMETERS=CTRL, DATA, indx
##  NEP
NEPDB_PARAMETERS=NEP, DATA, indx
##  SARM
SARMDB_PARAMETERS=SARM, SARM_DATA, sarm_index
##  TEMP ( Specify the temporary table space you want ASAP to use.)
TEMP_TS=TEMP

#############################################################################
## Configure ASAP Database User Password
#############################################################################
##For each ASAP server, type the password for each database schema. ASAP
database scripts create user schemas based on the user name and password that
you define.
##  ADM
ADMDB_PASSWORD=
##  Control
CTRLDB_PASSWORD=
##  NEP
NEPDB_PASSWORD=
##  SARM
SARMDB_PASSWORD=

#############################################################################
##  Configure ASAP UNIX Environment Variables - ASAP Server Port
#############################################################################
##Define the port number for ASAP servers.
##  CTRL Server
CTRL_PORT=30001
##  SARM Server
SARM_PORT=30002
##  NEP Server
NEP_PORT=30003
##  Define the port number for work order event notification for the OCA SRP.
This information will be populated in tbl_asap_srp table in the SARM
```

```
database.
## Oca Server
OCA_PORT=30004
##  Define the port number for the Java SRP server to send work orders. This
information will be populated in the tbl_listens table in the Control
database. Define the port number for the Java SRP server to receive work
order events. This information will be populated in the tbl_asap_srp table in
the SARM database.
##  JSRP - Sending WO
JSRPsend_PORT=30005
##   JSRPS123 - Receiving WO event
JSRPrecev_PORT=30006
##   ASAP DAEMON
DAEMON_PORT=30007
##   JeNEP Listener Port
NEP_Listener_PORT=30008

################################################################################
##  Configure Oracle WebLogic Server for ASAP
################################################################################
##  Path to the WebLogic installation directory
WL_HOME=
##  User Name of the Oracle WebLogic Server Administrator
WLS_USER=
##  HOST for the Oracle WebLogic Server
WLS_HOST=
##      Port for the Oracle WebLogic AdminServer
##  ----------------------------------------------
##  Specify NON-SSL Port if WLS_SSL_ENABLED is false
##  Specify SSL Port if WLS_SSL_ENABLED is true
WLS_ADMIN_PORT=
##  NON SSL Port for the target server of the WebLogic
WLS_TARGET_PORT=
##  SSL Port for the target server of the WebLogic
WLS_TARGET_SSL_PORT=
##  Oracle WebLogic Server
WLS_SERVER=AdminServer
## Use SSL?
WLS_SSL_ENABLED=false
## SSL KeyStore File
KEYSTORE_FILE=

################################################################################
## Configure Oracle WebLogic Server Passwords
################################################################################
##  Each password must be at least 8 characters long, and must contain at
least 1 number or special character.
##  ASAP_admin
ADMIN_PASSWORD=
##  cmws_studio
CMWS_PASSWORD=
##  ASAP_monitor
MONITOR_PASSWORD=
##  ASAP_operator
OPERATOR_PASSWORD=
##  WebLogic admin password
```

```
WLS_PASSWORD=
##  asap_ws_user
WS_PASSWORD=

#########################################################################
## Configure GRPC specific properties
#########################################################################
## SSL for GRPC ? 1 = Enabled, 0 = Not Enabled
GRPC_SSL_ENABLED=1
## Authentication for GRPC ? 1 = Enabled, 0 = Not Enabled
GRPC_AUTH_ENABLED=1
## Absolute path of SSL identity certificate in the pkcs12 format
SSL_IDENTITY_PKCS12_LOCATION=
## Absolute path of SSL trust certificate in the pkcs12 format
SSL_TRUST_PKCS12_LOCATION=
```

The various sections, their properties, and the description are explained in the following tables.

Configure ASAP UNIX Environment Variables:

Table B-1 lists the configuration parameters for ASAP Linux variables.

Table B-1 Configure ASAP UNIX Environment Variables

Name	Description	Required?
ORACLE_HOME	Enter the name of the base directory of the Oracle database client program. ASAP installation generates an UNIX environment variable(ORACLE_HOME) based on the directory name.	Yes
ANT_HOME	Specifies the location of ANT.	Yes
ASAP_SYS	Select the ASAP mode • Production (PROD) • Development (TEST) ASAP loads static provisioning configuration information from the database based on the mode. The default value is **TEST**.	Yes
ENV_ID	ASAP environment ID (maximum 4 alphanumeric characters).	Yes

Configure ASAP Database:

Table B-2 lists the parameters required for configuring ASAP database.

Table B-2 Configure ASAP Database

Name	Description	Required?
DSQUERY	Specifies the name of Oracle RDBMS Server. The ASAP installation program retrieves the name of the Oracle RDBMS Server from **tnsnames.ora** file on the installed Oracle client program.	Yes

Table B-2 (Cont.) Configure ASAP Database

Name	Description	Required?
DB_USER	Specifies the Oracle Server Database administrator User Name.	Yes
DB_PASSWORD	Specifies the Oracle Server Database administrator Password.	No (in Console mode)
CREATE_USERS	Specifies whether the Oracle Database user is created. When CREATE_USERS is set to **true** which is the default value, the database users are created during the ASAP configuration. When CREATE_USERS is set to **false**, you must have the database users already.	Yes
RACDB	Specifies whether this is an Oracle RAC database. Set to **true** for an Oracle RAC database. Set to **false** if it is not an Oracle RAC database. The default value is **false**.	Yes
RACDB_CONNECT_STRING	Specifies the value of the RAC Database Connection String. If RACDB is true, the value of the RAC Database Connection String is in the format:`Host1:Port1:ServiceName,Host2:Port2:ServiceName, and so on.`	Yes, if RACDB is true

Configure ASAP Database Table spaces:

Table B-3 lists the parameters to configure ASAP database table spaces.

Table B-3 Configure ASAP Database Table spaces

Name	Description	Required?
ADMDB_PARAMETERS	Specifies the required table space for each ASAP server. The recommended values are: `ADMDB_PARAMETERS=ADM, DATA, indx` Where • `ADM` is the ASAP schema. • `DATA` is the recommended data table space. • `indx` is the recommended index table space. ASAP database scripts populate the database schema based on the defined values.	Yes

Table B-3 (Cont.) Configure ASAP Database Table spaces

Name	Description	Required?
CTRLDB_PARAMETERS	Specifies the required table space for each ASAP server. The recommended values are: `CTRLDB_PARAMETERS=CTRL, DATA, indx` Where • `CTRL` is the ASAP schema. • `DATA` is the recommended data table space. • `indx` is the recommended index table space. ASAP database scripts populate the database schema based on the defined values.	Yes
NEPDB_PARAMETERS	Specifies the required table space for each ASAP server. The recommended values are: `NEPDB_PARAMETERS=NEP, DATA, indx` Where • `NEP` is the ASAP schema. • `DATA` is the recommended data table space. • `indx` is the recommended index table space. ASAP database scripts populate the database schema based on the defined values.	Yes
SARMDB_PARAMETERS	Specifies the required table space for each ASAP server. The recommended values are: `SARMDB_PARAMETERS=SARM, SARM_DATA, sarm_index` Where • `SARM` is the ASAP schema. • `SARM_DATA` is the recommended data table space. • `sarm_index` is the recommended index table space. ASAP database scripts populate the database schema based on the defined values.	Yes
TEMP_TS	Temporary table space that ASAP must use.	Yes

Configure ASAP Database User Password:

Table B-4 lists the parameters to configure ASAP database user password.

Table B-4 Configure ASAP Database User Password

Name	Description	Required?
ADMDB_PASSWORD	Specifies password for each database schema. For each ASAP server, type the password for each database schema. ASAP database scripts create user schemas based on the user name and password that you define.	Yes (in Silent Mode)

Table B-4 (Cont.) Configure ASAP Database User Password

Name	Description	Required?
CTRLDB_PASSWORD	Specifies password for each database schema. For each ASAP server, type the password for each database schema. ASAP database scripts create user schemas based on the user name and password that you define.	Yes (in Silent Mode)
NEPDB_PASSWORD	Specifies password for each database schema. For each ASAP server, type the password for each database schema. ASAP database scripts create user schemas based on the user name and password that you define.	Yes (in Silent Mode)
SARMDB_PASSWORD	Specifies password for each database schema. For each ASAP server, type the password for each database schema. ASAP database scripts create user schemas based on the user name and password that you define.	Yes (in Silent Mode)

Configure Oracle WebLogic Server for ASAP:

Table B-5 lists the parameters to configure Oracle WebLogic server for ASAP.

Table B-5 Configure Oracle WebLogic Server for ASAP

Name	Description	Required?
WL_HOME	Specifies the path to the WebLogic installation directory.	Yes
WLS_HOST	Specifies the host for the Oracle WebLogic Server.	Yes
WLS_USER	Specifies the username of the Oracle WebLogic Server Administrator.	Yes
WLS_ADMIN_PORT	Specify NON-SSL port if WLS_SSL_ENABLED is false. Specify SSL port if WLS_SSL_ENABLED is true.	Yes
WLS_TARGET_PORT	Specifies the non-SSL port for the target server of the WebLogic.	Yes
WLS_TARGET_SSL_PORT	Specifies the SSL port for the target server of the WebLogic.	Yes if SSL is enabled
WLS_SERVER	Specifies the Oracle WebLogic server. The default value is **AdminServer**.	Yes
WLS_SSL_ENABLED	Set to **true** if SSL is enable. The default value is **false**.	Yes
KEYSTORE_FILE	Specifies the path of the SSL KeyStore file.	Yes if SSL is enabled

ASAP Server Ports:

Table B-6 lists the parameters to configure ASAP server ports.

Table B-6 ASAP Server Ports

Name	Description	Required?
SARM_PORT	Specifies the port number of the SARM server. The default value is 30002.	Yes
CTRL_PORT	Specifies the port number of the control server. The default value is 30001.	Yes
NEP_PORT	Specifies the port number of the NEP server. The default value is 30003.	Yes
OCA_PORT	Specifies the port number for work order event notification for the OCA SRP. The default value is 30004.	Yes
JSRPsend_PORT	Specifies the port number for the Java SRP server to send work orders. The default value is 30005.	Yes
JSRPrecev_PORT	Specifies the port number for the Java SRP server to receive work order events. The default value is 30006.	Yes
DAEMON_PORT	Specifies the port number of the daemon server. The default value is 30007.	Yes
NEP_Listener_PORT	Specifies the port number of the NEP listener. The default value is 30008.	Yes

Configure WebLogic Passwords for Silent mode:

Table B-7 lists the parameters to configure passwords for installing ASAP in the silent mode.

Table B-7 Configure WebLogic Passwords for Silent mode

Property	Description	Required?
ADMIN_PASSWORD	Specifies the password for ASAP_admin.	No (in Console mode)
CMWS_PASSWORD	Specifies the password for CMWS Studio.	No (in Console mode)
MONITOR_PASSWORD	Specifies the password for the ASAP_Monitor.	No (in Console mode)
OPERATOR_PASSWORD	Specifies the password for the ASAP_operator.	No (in Console mode)
WLS_PASSWORD	Specifies the password for the WebLogic admin.	No (in Console mode)
WS_PASSWORD	Specifies the password for the ASAP_ws_user.	No (in Console mode)

Configure GRPC specific properties:

Table B-8 lists the parameters to configure GRPC specific properties.

Table B-8 Configure GRPC specific properties

Property	Description	Required?
GRPC_SSL_ENABLED	Specifies if SSL is enabled for GRPC. The valid values are: • 1 - Enabled • 0 - Disabled	No
GRPC_AUTH_ENABLED	Specifies if the authentication is enabled for GRPC. The valid values are: • 1 - Enabled • 0 - Disabled	No
SSL_IDENTITY_PKCS12_LOCATION	Specifies the absolute path of SSL identity certificate in the pkcs12 format.	Yes if GRPC_SSL_ENABLED is enabled
SSL_TRUST_PKCS12_LOCATION	Specifies the absolute path of SSL trust certificate in the pkcs12 format.	Yes if GRPC_SSL_ENABLED is enabled