SAP NetWeaver® Application Server ABAP/Java on Oracle Cloud Infrastructure

ORACLE WHITE PAPER | SEPTEMBER 2018

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Revision History

The following revisions have been made to this white paper since its initial publication:

Date	Revision
September 3, 2018	Updated link for metrics collector.
August 3, 2018	Added information about a new block storage size.
June 22, 2018	Added information about supported Oracle Cloud Infrastructure Compute virtual machine shape types and an example for installation on a VM.DenseIO2.16 shape.
May 21, 2018	Added information about support for Oracle Cloud Infrastructure Compute X7 shapes.

You can find the most recent versions of the Oracle Cloud Infrastructure white papers at https://cloud.oracle.com/iaas/technical-resources.

Table of Contents

Purpose of This White Paper

This technical white paper is a reference guide for deploying SAP NetWeaver® Application Server ABAP/Java on Oracle Cloud Infrastructure, following suggested platform best practices. It also discusses details about combining parts of Oracle Cloud Infrastructure, Oracle Linux, Oracle Database instances, and SAP application instances to run software products based on SAP NetWeaver® Application Server ABAP/Java in Oracle Cloud Infrastructure.

This white paper is not a full reference for SAP NetWeaver® Application Server ABAP/Java. Rather, it describes how to plan and implement an SAP landscape in the cloud in a supported and verified way.

Assumptions

This white paper assumes the following knowledge:

- You are familiar with the fundamentals of Oracle Cloud Infrastructure. For information, see the Oracle Cloud Infrastructure technical documentation.
- You have a background in SAP NetWeaver® Application Server ABAP/Java using Oracle Database and Oracle Linux. For more information, see the following resources:
  - http://go.sap.com/solution.html
  - https://www.sap.com/community/topic/oracle.html
  - http://docs.oracle.com/en/operating-systems/linux.html
- You are familiar with the documentation for the following products:
  - Oracle Cloud Infrastructure
  - Oracle Database 11g and 12c
  - Oracle Linux 6 and 7
  - SAP NetWeaver® 7.*x*

Most of the steps described here are the same as in a traditional SAP deployment in a customer data center. The document also includes details about how to develop a backup and high-availability plan for your SAP installation in Oracle Cloud Infrastructure.

Overview of Oracle Cloud Infrastructure

Oracle Cloud Infrastructure offers a set of core infrastructure capabilities, like compute and storage, that enable customers to run any workload in the cloud. It also offers a comprehensive set of integrated, subscription-based, infrastructure services that enable businesses to run any

workload in an enterprise-grade cloud that is managed, hosted, and supported by Oracle. Two types of infrastructure as a service (IaaS) are currently offered: Oracle Cloud Infrastructure Classic and Oracle Cloud Infrastructure. Only Oracle Cloud Infrastructure has been certified and is offered for SAP NetWeaver® Application Server ABAP/Java. Oracle Cloud Infrastructure combines the elasticity and utility of public cloud with the granular control, security, and predictability of on-premises infrastructure to deliver high-performance, high-availability, and cost-effective infrastructure services.

Regions and Availability Domains

Oracle Cloud Infrastructure is physically hosted in regions and availability domains. A *region* is a localized geographic area, and an *availability domain* is one or more data centers located within a region. A region is composed of several availability domains. Most Oracle Cloud Infrastructure resources are either region-specific, such as a Virtual Cloud Network, or availability domain–specific, such as a compute instance.

Availability domains are isolated from each other, fault tolerant, and unlikely to fail simultaneously. Because availability domains do not share infrastructure such as power or cooling, or the internal availability domain network, a failure at one availability domain is unlikely to impact the availability of the others.

All the availability domains in a region are connected to each other by a low-latency, high-bandwidth network. This connection makes it possible to provide high-availability connectivity to the internet and customer premises, and to build replicated systems in multiple availability domains for both high availability and disaster recovery

Regions are completely independent of other regions and can be separated by vast distances—across countries or even continents. Generally, an application should be deployed in the region where it is most heavily used, because using nearby resources is faster than using distant resources.

For an SAP NetWeaver environment, all the components of an SAP NetWeaver system (such as Dialog Instances, Central Instance, Central Services, Web Dispatcher, Gateway or SAP Database) must be within the same region. For performance reasons, these components should be within the same availability domain.

Hybrid deployments between on-premises and cloud are not supported because of network latency.

Services

Oracle Cloud Infrastructure offers the following services.

Identity and Access Management

Oracle Cloud Infrastructure provides Identity and Access Management (IAM) at no additional cost. IAM lets you control who has access to your cloud resources and what type of access they have. You can manage complex organizations and rules with logical groups of users and resources, and define policies. IAM helps you to set up administrators, users, and groups, and to specify their permissions. It allows you to use a single model for authentication and authorization to securely control access and easily manage your IT resources across all Oracle Cloud Infrastructure.

Networking

Oracle Cloud Infrastructure Networking helps you set up virtual versions of traditional network components. Networking is the cornerstone of any cloud platform. It defines performance and the customer experience. Extend your IT infrastructure with highly customizable virtual cloud networks (VCNs) and connectivity services that offer predictable and consistent performance, isolation, and availability.

A VCN is a customizable and private network in Oracle Cloud Infrastructure. Just like a traditional data center network, the VCN provides you with complete control over your network environment. You can assign your own private IP address space, create subnets and route tables, and configure security lists (stateful firewalls). A single tenant can have multiple VCNs, thereby providing grouping and isolation of related resources.

Oracle Cloud Infrastructure FastConnect is a network connectivity alternative to using the public internet for connecting your network with Oracle Cloud Infrastructure. FastConnect provides an easy, elastic, and economical way to create a dedicated and private connection with higher bandwidth options, and it provides a more reliable and consistent networking experience when compared to internet-based connections.

Oracle Cloud Infrastructure's flat and fast network provides the latency and throughput of rack adjacency across the whole network, which allows synchronous replication and constant uptime. No network oversubscription also provides predictable bandwidth and performance.

Compute

Oracle Cloud Infrastructure Compute helps you provision and manage compute hosts, known as compute instances, to meet your compute and application requirements. Multiple compute options provide the flexibility to run your most demanding workloads, as well as less compute-intensive applications, in a secure and highly available cloud environment.

Oracle's approach includes options for local storage for Oracle Cloud Infrastructure compute instances, enabling solutions that require high IOPS and low latency. They provide industry-first, fully dedicated, bare metal servers on a software-defined network, and virtual machine servers with dedicated CPU, memory, disk and network resources. Compute offers unrivaled performance with up to 52 processor cores on bare metal, up to 24 dedicated processor cores on virtual machine servers, and the latest Non-Volatile Memory Express (NVMe) SSDs providing millions of IOPS. Bare metal and virtualized servers with a “VM. DenseIO2”-class shape type are ideal for I/O intensive applications or big data workloads, and are the best environment for running Oracle Databases and SAP systems. As virtual servers do not make use of CPU-, memory-, disk- or network overprovisioning their performance is predictable and constant.

Block Volumes

Oracle Cloud Infrastructure Block Volumes helps you dynamically provision and manage block storage volumes. This service provides high-speed storage capacity with seamless data protection and recovery. Network-attached block volumes deliver low latency and tens of thousands of IOPS per compute instance, which allows you to improve the availability, performance, and security of your applications, and increase your customer service levels.

Object Storage

Oracle Cloud Infrastructure Object Storage helps you manage data as objects stored in containers. Object Storage offers an unlimited amount of capacity, automatically replicating and healing data across multiple fault domains for high durability and data integrity. You can enhance the scale and performance of content-rich, analytic, and backup applications to serve more customers and achieve results faster.

Overview and Architecture of SAP NetWeaver Application Server ABAP/Java

SAP NetWeaver® Application Server ABAP/Java is a major application platform of SAP SE. It is the technical foundation for many SAP applications. The SAP NetWeaver® Application Server is the runtime environment for many SAP applications, such as the SAP Business Suite.

The SAP NetWeaver® Application Server ABAP is a main building block of SAP's software stack. The SAP NetWeaver® ABAP/Java platform is designed as a three-tier architecture with a presentation layer (SAPGUI, browser), an application layer (AS ABAP, AS Java), and a database layer. These three layers can run on different computers. If the application layer and the database layer run on the same computer, then this topology is called an SAP two-tier setup.

SAP NetWeaver® Application Server ABAP/Java forms the application platform for all SAP products and industry solutions written in ABAP (such as SAP ERP, SAP CRM, SAP SRM, and SAP BW) and Java (such as SAP Portal and SAP PI).

Design

The SAP NetWeaver® Application Server is designed to provide a robust and supportable architecture for the SAP applications and solutions running on it. The SAP NetWeaver® Application Server consists of Application Server ABAP (AS ABAP) and Application Server Java (AS Java).

Application Server ABAP

Application Server ABAP provides the complete technology and infrastructure to run ABAP applications. The kernel of AS ABAP is written in C/C++.

Application Server Java

Application Server Java provides a Java™ 2 Enterprise Edition (Java EE) 1.5 compliant environment for developing and running Java EE programs.

Oracle Database

ABAP programs access the database through the database interface of AS ABAP, which is subdivided into an Open SQL interface and a native SQL interface. Open SQL is a subset of the Structured Query Language (SQL) realized directly by ABAP statements. Native SQL consists of database-specific SQL instructions that are passed directly to the database system (either statically or dynamically via ADBC). The database interface of AS ABAP for the Oracle Database uses the Oracle Call Interface (OCI).

Java programs access the database through the database interface of AS Java. The database interface for AS Java uses the Oracle thin JDBC driver.

Technical Components

An SAP system consists of several application server instances and one database system. In addition to multiple dialog instances, the System Central Services (SCS) for AS Java instance and the ABAP System Central Services (ASCS) for AS ABAP instance provide message server and enqueue server for both stacks.

A dialog instance with AS ABAP and AS Java consists of the following components:

- The Internet Communication Manager (ICM) sets up the connection to the internet. It can process both server and client web requests. The SAP NetWeaver® Application Server can act as a web server or a web client.
- Central services (message server and enqueue server) are used for lock administration, message exchange, and load balancing in the SAP system.
- AS ABAP components (on the left side in the following graphic):
    - The dispatcher distributes the requests to the work processes. If all the processes are occupied, the requests are stored in the dispatcher queue.
    - The work processes execute ABAP or Java programs.
    - The SAP Gateway provides the RFC interface between the SAP instances (within an SAP system and beyond system boundaries).
- AS Java components (on the right side in the following graphic):
    - The Server Processes execute Java requests.
    - The instance controller controls and monitors the life cycle of the AS Java instance.

The following graphic gives an overview of the components of the SAP NetWeaver® Application Server:

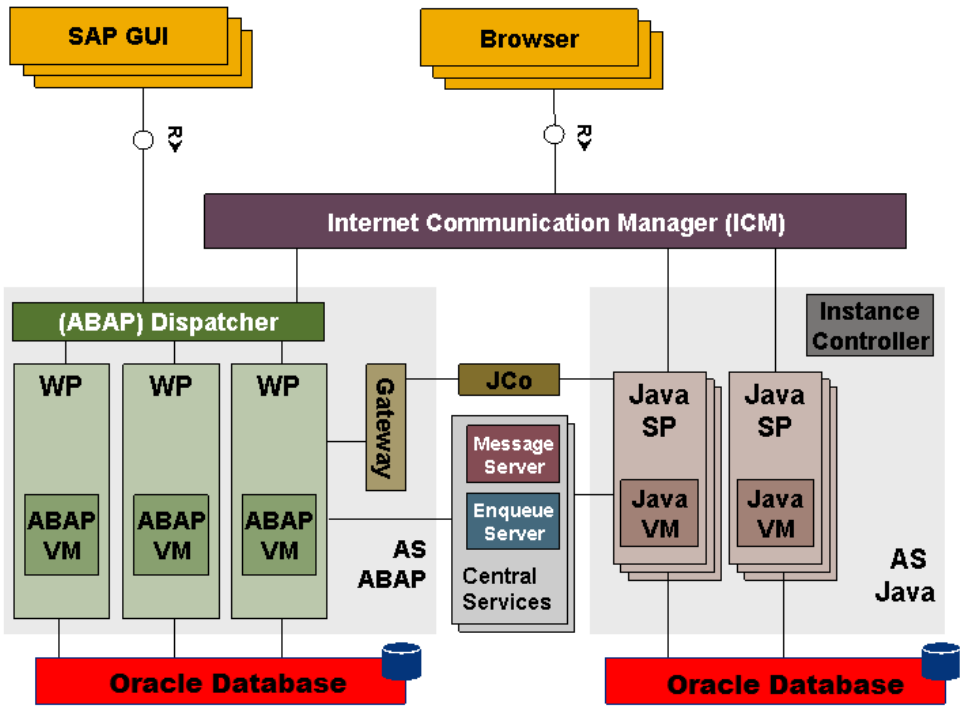

Figure 1: Components of the SAP NetWeaver® Application Server

Overview of SAP NetWeaver® Application Server ABAP/Java on Oracle Cloud Infrastructure

Oracle Cloud Infrastructure offers hourly and monthly metered bare metal and virtual machine compute instances with up to 51.2 TB of locally attached NVMe SSD storage or up to 1 PB of iSCSI attached block storage. A 28-TB NVMe instance is capable of over 3 million 4K IOPS, the ideal platform for an SAP NetWeaver® workload using an Oracle Database.

Instances in the Oracle Cloud Infrastructure are attached using a 10 or 25 Gbps non-blocking network with no oversubscription. While each compute instance running on bare metal has access to the full performance of the interface, virtual machine servers can rely on guaranteed network bandwidths and latencies; there are no “noisy neighbors” to share resources or network bandwidth with. Compute instances in the same region are always less than 1 ms away from each other, which means that your SAP application transactions will be processed in less time, at a lower cost than with any other IaaS provider.

To support highly available SAP deployments, Oracle Cloud Infrastructure builds regions with at least three availability domains. Each availability domain is a fully independent data center with no fault domains shared across availability domains. An SAP NetWeaver® Application Server ABAP/Java landscape can span across multiple availability domains.

Recommended Instances and Topologies for SAP NetWeaver® Application Server ABAP/Java Installation

This section describes the recommended options for installing SAP NetWeaver® Application Server ABAP/Java on Oracle Cloud Infrastructure.

SAP Application Tier

You can use the following Oracle Cloud Infrastructure Compute instance shapes to run the SAP application.

Bare Metal Compute

- BM.Standard1.36
- BM.DenseIO1.36
- BM.Standard2.52
- BM.DenseIO2.52

These compute instance shapes have 36 or 52 CPU cores and a root volume size of ~50 GB by default. We strongly recommend that you do not use the root volume as the destination for SAP NetWeaver® software. Instead, use NVMe storage if it's available, or attach a separate block volume to the instance that is big enough to run SAP instances to support a very large-scale deployment**.**

Virtual Machine Compute

- VM.Standard2.1
- VM.Standard2.2
- VM.Standard2.4
- VM.Standard2.8
- VM.Standard2.16
- VM.DenseIO2.8
- VM.DenseIO2.16

These compute instance shapes have 1,2,4,8 or 16 CPU cores and a root volume size of ~50 GB by default. We strongly recommend that you do not use the root volume as the destination for SAP NetWeaver® software. Instead, use NVMe storage if it's available or attach a separate block volume to the instance that is big enough to run SAP instances to support a large-scale deployment.

SAP Database Tier

You can use the following Oracle Cloud Infrastructure Compute instance shapes for the SAP database tier running the Oracle Database.

Bare Metal Compute

- BM.Standard1.36
- BM.DenseIO1.36
- BM.Standard2.52
- BM.DenseIO2.52

Virtual Machine Compute

- VM.Standard2.1
- VM.Standard2.2
- VM.Standard2.4
- VM.Standard2.8
- VM.Standard2.16
- VM.DenseIO2.8
- VM.DenseIO2.16

We strongly recommend protecting the database by using Oracle Data Guard between two availability domains.

Topologies of SAP NetWeaver® Application Server ABAP/Java on Oracle Cloud Infrastructure

There are various installation options for SAP NetWeaver® Application Server ABAP/Java. You can place one complete SAP application layer and the Oracle Database on a single compute instance (two-tier SAP deployment). You can install the SAP application layer instance and the database instance on two different compute instances (three-tier SAP deployment). Based on the sizing of your SAP systems, you can deploy multiple SAP systems on one compute instance in a

two-tier way or distribute those across multiple compute instances in two-tier or three-tier configurations. To scale a single SAP system, you can configure additional SAP dialog instances (DI) on additional compute instances.

A key element of the installation is a bastion host with access to the network where the other compute instances are located and access from outside is managed. A bastion host can have the following roles:

- Provide a VNC server for graphical access and an SSH server from outside
- Deliver a graphical workspace for any related operations (for example, download, install, and access)
- Work as an NFS server to provide SAP installation media, SAP patches, and SAP Bundle Patches for the Oracle Database
- Work as an NFS server for the shared SAP file systems `/sapmnt` and `/usr/sap/trans`
- Work as a ULN Proxy to provide operating system updates for Oracle Linux, without registration of all the compute instances

Planning Your SAP Implementation

This section provides guidance for planning your implementation of SAP NetWeaver® Application Server ABAP/Java on Oracle Cloud Infrastructure.

Instance Model

The only instances certified and supported for SAP NetWeaver® Application Server ABAP/Java installations on Oracle Cloud Infrastructure are Oracle Linux 6 and Oracle Linux 7 with Oracle Database 11g Release 2 (11.2.0.4), Oracle Database 12c Release 1 (12.1.0.2), or Oracle Database 12c Release 2 (12.2.0.1).

The following restrictions apply:

- Only Unicode deployments of SAP NetWeaver® Application Server ABAP/Java are supported.
- Only Oracle Database single instance installations on file systems are supported.
- There is no support for Oracle Automatic Storage Management (ASM) on compute instances.
- No hybrid deployment between on-premises and cloud are supported because of network latency.

- Only the hypervisor and virtualization technology used by Oracle Cloud Infrastructure Virtual Machines is supported and certified for Oracle Cloud Infrastructure Compute instances.

Licenses

If you have already bought Oracle Database licenses from SAP (ASFU), you can transfer them to Oracle Cloud Infrastructure. Notify SAP that you intend to bring your own license (BYOL).

The same applies for licenses that you have bought from Oracle (Full Use, FU). If you have enough licenses, you can also transfer them from on-premises to Oracle Cloud Infrastructure. To ensure that the number of shapes, processors, and cores is correct, we recommend that you check with your Oracle sales manager or local license sales contact. They will help you to get the correct licensing in place.

Support

If you need technical support or help with Oracle Cloud Infrastructure, you can go to My Oracle Support and create a service request. If you encounter any problem with the SAP NetWeaver® Application Server ABAP/Java deployment with Oracle Cloud Infrastructure, open a support message with SAP support and assign it to the support queue BC-OP-LNX-OLNX.

Customers must purchase the Oracle Cloud Infrastructure service directly from Oracle to use Oracle Cloud Infrastructure and get support for it. Details about the service provided to the customer are described in the document Oracle Cloud Hosting and Delivery Policies.

In addition to support for technical issues, use My Oracle Support if you need to perform the following tasks:

- Reset the password or unlock the account for the tenancy administrator
- Add or change a tenancy administrator
- Request a service limit increase

Note: SAP Note 2520061 - SAP on Oracle Cloud Infrastructure: Support prerequisites describes the support subscriptions that are needed to run SAP NetWeaver® Application Server ABAP/Java on Oracle Cloud Infrastructure with Oracle Linux.

Documentation

Determine the supported combination for Oracle Linux and Oracle Database for your planned SAP product by using the SAP Product Availability Matrix (PAM). The SAP PAM points to the relevant

SAP NetWeaver® installation guides. Ensure that you are familiar with the relevant SAP NetWeaver® master and installation guides and the referenced SAP notes within. To find planning, installation, patching, and operation documentation for your task, see the SAP NetWeaver® Guide Finder.

Become familiar with product documentation for any components of your stack: Oracle Cloud Infrastructure, Oracle Linux, Oracle Database, and SAP NetWeaver® Application Server ABAP/Java.

Note: SAP Note 2474949 - SAP NetWeaver® on Oracle Cloud Infrastructure defines all the technical prerequisites for deploying an SAP NetWeaver® Application Server ABAP/Java system with Oracle Cloud Infrastructure. This note is updated regularly, so read it before you start any deployment. Information in the note takes precedence over information in this paper.

Workload Size

Estimate the needed size for your SAP installation with the SAP Quick Sizer tool, and calculate the needed Oracle Cloud Infrastructure Compute instances for your SAP workload. For SAPS numbers, you can also consult SAP Note 2474949, although the SAPS numbers listed there are only for a performance indication and have not been achieved by using a high-performance benchmark.

Shape	Instance Type	OCPU	Memory	Storage (Database Storage)
BM.Standard1.36	Standard compute capacity	36	256	Block storage only (1 PB raw)
BM.DenseIO1.36	Dense I/O compute capacity	36	512	9 x 3.2 TB NVMe devices (28.8 TB raw)
BM.Standard2.52	X7-based standard compute capacity	52	768	Block storage only (512 TB raw)
BM.DenseIO2.52	X7-based dense I/O compute capacity	52	768	8 x 6.4 TB NVMe devices (51.2 TB raw)
VM.DenseIO2.16	X7-based dense I/O compute capacity	16	240	2 x 6.4 TB NVMe devices (12.8 TB raw)
VM.DenseIO2.8	X7-based dense I/O compute capacity	8	120	1 x 6.4 TB NVMe devices
VM.StandardIO2.16	X7-based standard I/O compute capacity	16	240	Block storage only (1 PB raw)
VM.StandardIO2.8	X7-based standard I/O compute capacity	8	120	Block storage only (1 PB raw)

Shape	Instance Type	OCPU	Memory	Storage (Database Storage)
VM.StandardIO2.4	X7-based standard I/O compute capacity	4	60	Block storage only (1 PB raw)
VM.StandardIO2.2	X7-based standard I/O compute capacity	2	30	Block storage only (1 PB raw)
VM.StandardIO2.1	X7-based standard I/O compute capacity	1	15	Block storage only (1 PB raw)

Note: Block storage can be attached to each shape type.

Presales and consulting teams from Oracle can help you determine valid sizing for your planned SAP landscape in the cloud.

Planning the SAP Deployment

Use the information in this section to plan your SAP NetWeaver® Application Server ABAP/Java deployment on Oracle Cloud Infrastructure.

Network

Initially, you have to set up a VCN where you can define different subnets. You must define a VCN before you can create other resources (for example, compute instances).

A VCN is specific to a region. After creating a VCN, you can add one or more subnets in each availability domain. A specific Classless Inter-Domain Routing (CIDR) block is specified for each subnet and must be a subset of the VCN. For more information, see Overview of Networking in the Oracle Cloud Infrastructure documentation.

Security can be configured at several levels within a VCN. A subnet can be designated as public or private. A private subnet cannot have a public IP address. Security lists can control packet-level traffic into and out of a subnet or an instance. In addition, at the instance level, firewall rules can be implemented. Gateways and route tables provide control over traffic flow between the VCN and outside destinations. Finally, IAM polices provide control over who can access and configure which resources.

For naming, each subnet can resolve names to the internet or within a VCN. In addition, an on-premises DNS server can be added to the search scope. A description of the choices for using DNS in your VCN are described in the Oracle Cloud Infrastructure Networking documentation.

Maintaining accurate time is a key requirement for maintaining secure communications because the current time is used as an ingredient for encrypting data. Oracle Cloud Infrastructure provides a private NTP server without the need for a dedicated connection to the internet. It is crucial to have the correct time in an SAP system and the database system so your compute instances are always synchronized. All compute instances of an SAP system must be in the same time zone.

You have various choices of where to put the compute instances. Following are some possible scenarios:

- Separation of public subnet, management private subnet, and apps and database private subnet
- Same separation as above but also a different private subnet for apps and databases
- Separation of different SAP landscapes in different VCNs
- Separation into test, quality, and production VCNs
- Migration of your existing on-premises network to the cloud

A local firewall for each compute instance that comes from the operating system, and security lists that are part of the Oracle Cloud Infrastructure Networking service, allow and deny specific network traffic. For an SAP deployment, the local firewall must be disabled, and only the security lists for the subnets must be managed. You can get an overview about the required ports for an SAP system from SAP Help Ports**.**

Storage

The database files of Oracle Databases are supported only on file systems. All files belonging to a database need to be protected.

Follow the Oracle Cloud Infrastructure documentation, Protecting Data on NVMe Devices, and the Oracle Linux administration guides to protect your data located on NVMe.

Block Storage can be attached as well, and is supported for database files. However, for better performance and throughput, we recommend using NVMe storage.

Compute Instances

Oracle provides different images or a template of a virtual hard drive for the compute instances. Those images determine the operating system. For SAP NetWeaver® Application Server ABAP/Java installation, only the images for Oracle Linux 6 and Oracle Linux 7 are supported.

Implementing Your Plan

This section provides the steps for implementing your planned deployment of SAP NetWeaver® Application Server ABAP/Java in Oracle Cloud Infrastructure.

Get Your Oracle Cloud Infrastructure Account

To get your Oracle Cloud Infrastructure account, work with your Oracle account team. They can help you find the best option for structuring your subscription. Options include metered, nonmetered, and trial subscriptions. A convenient way to start immediately is to sign up for a free trial by using the instructions listed in Signing Up for Oracle Cloud Infrastructure.

Prepare Your Environment

You can set up all resources by using the Oracle Cloud Infrastructure Console or by using automation. Automation provides the advantage of repeatability, while the Oracle Cloud Infrastructure Console provides immediate provisioning and a human-friendly user interface.

Set Up the Bastion Host

Oracle recommends that you use Oracle Linux 7 on the bastion host.

Set Up the ULN Proxy

To ensure that you have the latest operating system updates for Oracle Linux 6 and 7 available from Oracle, register the system with Oracle Unbreakable Linux Network (ULN) and set up a ULN proxy. A proxy enables you to update compute instances with the latest packages even if the compute instance is not connected to the internet. A requirement for maintaining the proxy is to ensure that sufficient disk space is available to hold all the updates.

Register your Oracle Linux 7 system to ULN and follow the ULN Users Guide to configure a ULN proxy to mirror the needed local channels. Provide a block volume after you approximate the size of your needed channels.

Set Up the NFS Server

When the bastion host is configured as an NFS server, installation media can be shared securely with other compute instances that do not have internet access. When configuring the NFS server, consider the amount of disk space needed and the security rule configuration.

Configure an NFS server on the bastion host and follow the description in the Oracle Linux Administration Guide for Release 7. Define directories for installation media, updates, and the shared SAP file systems after you create and attach the block volumes.

Set Up the VNC Server

GUI access at the operating-system level is needed to run any graphical tools. The native GUI can be accessed by enabling a VNC server on the bastion host. Ensure that security lists are maintained to allow access to only approved sources.

Configure a VNC server on the bastion host and follow the description in the Oracle Linux Administration Guide for Release 7. Implement local firewall rules or entries into the security lists to allow access to the VCN from your network outside.

Set Up the SAP Download Manager

SAP Download Manager helps you download software from the SAP Software Download Center (SWDC) that you have put in the download basket. Install the SAP Download Manager on the bastion host and set the needed S-User and password credentials to download SAP software from the SWDC.

Download Your SAP Software

From the SWDC, download the needed installation software for your specific SAP product. With your S-User permissions, you can download the installation media directly or you can use the SAP Download Manager. We recommend storing the software on a shared file system.

Configure the SAP GUI

Install and configure the SAP GUI for Java on the bastion host that is running Oracle Linux. With the unified SAP front end, you can connect to SAP NetWeaver® ABAP installations. Details are described in SAP Note 146505 and on the SAP Community Wiki.

SAP GUI for Java needs configuration information about your SAP environment, such as the names and addresses of your SAP servers. Based on this information, a connection directory is created that contains all available connections that can be selected in the SAP logon list. This directory can be centrally stored on a web server, and only a URL needs to be configured in SAP GUI for Java. Preset configuration and options can be distributed as templates during the initial installation process, so that a manual configuration after a first installation of SAP GUI for Java is not required. Access to the SAP ports for the connection needs to be created in the security lists.

Install SAP NetWeaver® Application Server ABAP/Java

This section describes the steps for installing SAP NetWeaver® Application Server ABAP/Java. An example that outlines the steps required for preparation and installation on a VM.DenseIO2.16 shape can be found at the end of this chapter.

Prepare the Operating System

SAP NetWeaver® Application Server ABAP/Java is certified to run on Oracle Cloud Infrastructure compute instances that are running Oracle Linux 6 and Oracle Linux 7.

Check that the following requirements for your Oracle Cloud Infrastructure compute instance are implemented. For detailed requirements for Oracle Linux 6, see SAP Note 1635808. For detailed requirements for Oracle Linux 7, see SAP Note 2069760.

- Install all needed RPMs for SAP NetWeaver® Application Server ABAP/Java and Oracle Database.
- Set SELinux to permissive mode.
- Prepare the system with the needed Linux kernel parameters, and set the process resource limits.
- Set the hostname.
- Set the needed information for NTP and DNS.

You need to increase the size of the swap space to the SAP recommended value provided in SAP Note 1597355. For the implementation, follow the specific Oracle Linux administration guide.

Verify that Transparent Huge Pages (THP) are disabled in cloud instances where databases are running. For details, see SAP Note 1871318.

Provision SAP Monitoring

For every cloud solution, SAP requires the collection of configuration and performance data for the cloud platform being used.

On bare metal and virtual machine compute nodes, the SAP Host Agent must be installed. Installation of the SAP Host Agent can be done either using the SAP Software Provisioning Manager (SWPM) or manually, as described in the ”SAP Host Agent Installation” topic in the SAP documentation.

The required version and patch level of the SAP Host Agent is 7.21 PL35 or higher. See SAP Note 2655715 for more details.

For virtual machine compute nodes, additional monitoring components must be installed to enable SAP enhanced monitoring.

The SAP Host Agent consumes configuration and performance metrics. These metrics are collected by a Linux service called oci-sap-metrics-collector. The oci-sap-metrics-collector service must be installed and started on each virtual machine compute node. It is shipped as an Linux RPM called oci-sap-metrics-collector-1.0-8.noarch.rpm.

Part of the package is a Python script that ensures that updates of the package are applied automatically and that the service is being started if it's not running. For this script, an entry in crontab of OS user root is created during installation of the package.

1. Download and install the package as user root:

```
https://objectstorage.eu-frankfurt-
1.oraclecloud.com/p/Ej8hZlFthynybW3Fi6UjcpKTJfVLMNwAP9wGyMH9GhU/n/imagegen
/b/metrics-collector-binary-store/o/oci-sap-metrics-collector-1.0-
8.noarch.rpm

yum -y install oci-sap-metrics-collector-1.0-8.noarch.rpm
```

2. Verify that metrics collection works as expected. As the root user, run the following command:

```
curl http://127.0.0.1:18181
```

This should return the XML document for consumption by the SAP Host Agent.

Logs for oci-sap-metrics-collector are written to /var/log/oci-sap-metrics.

Configure Storage

For an SAP deployment, you need to configure the following file systems:

- A `/usr/sap` directory for the SAP installation
- An `/oracle` directory for the Oracle Database
- Enough swap space according to the SAP recommendation

Examples of how to protect your data on the NVMe storage by using software raid are described in the Protecting Data on NVMe Devices topic of the Oracle Cloud Infrastructure documentation. Oracle recommends that the database files and the redo logs are mirrored on separated file systems. Other data should also be mirrored, based on the availability requirements of your installation. Recommended file system types are ext3, ext4 for Oracle Linux 6, and XFS for Oracle Linux 7.

Install SAP Software

After you create your recommended instance, finish the needed preparation of the operating system, prepare the SAP installation software and the needed SAP file system structure following the specific SAP NetWeaver® installation guide, and perform the installation with the latest available version of the SAP Software Provisioning Manager (SWPM).

Example: Prepare a VM.DenseIO2.16 shape Running Oracle Linux 6

All OS-specific configuration steps must be done as the user root.

1. Update the OS and install required RPMs:

```
yum -y update
yum -y groupinstall "Base" "Compatibility Libraries" "Debugging Tools"
"Directory Client" "Hardware Monitoring Utilities" "Large Systems
Performance" "Perl support" "Storage Availability Tools" "X window system"
"Development tools"

yum -y install tigervnc-server liberation-mono-fonts gnome-session gnome-
terminal gnome-screensaver gnome-panel compat-libstdc++-33 compat-libcap1
libaio-devel ksh uuidd vim parted xorg-x11-xauth xclock

yum -y install oracle-database-server-12cR2-preinstall.x86_64
```

2. Turn off and disable the local firewall:

```
service iptables stop ; chkconfig iptables off
```

3. Turn on and enable uuidd:

```
service uuidd start ; chkconfig uuidd on
```

4. Set the root password to allow you to log in as root when running SWPM.

   Run `passwd` and choose a password that will be accepted by the Linux password complexity check and by SWPM as the master password for all accounts being created. You can change them later according to your needs.

```
passwd root (e.g. $$99SapSapR3)
```

5. Prepare for NFS mounting of the volume with the installation media on the target host.
   - On Oracle Cloud Infrastructure: Add a security list to the firewall to allow NFS-specific communication between known servers.

- On the NFS server (where you have your installation media): Add the target host's IP address to the `/etc/exports` directory of the NFS server:

```
/mnt/vol1 <clientip>(rw,async,no_acl,no_root_squash)
service nfs reload
```

- On the NFS client: Create the target directory for the NFS mount:

```
mkdir /mnt/vol1
Add entry in /etc/fstab of nfs client
<serverip>:/mnt/vol1       /mnt/vol1       nfs
defaults,bg,_netdev,clientaddr=<clientip> 0 0
```

Note that `clientaddr=<clientip>` is required to select between the public or local IP address to be used for NFS.

6. Prepare the NVMe disks, the filesystem, and `fstab`:

A. Create the array

```
mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/nvme0n1
/dev/nvme1n1
```

Note that in this example, a RAID 1 is created for higher data protection. Virtual machine compute nodes of a VM.DenseIO2 shape type other than VM.DenseIO2.16 might have only one NVMe disk. Virtual machine compute nodes of shape types VM.StandardIO2 do not have any NVMe disks, and you have to attach block storage devices first.

B. Create the filesystem:

```
mkfs -t ext4 /dev/md0
```

C. Create a directory for mounting the array:

```
mkdir /disk1
```

D. Determine the UUID of the array and add it to `/etc/fstab` to mount it.

```
blkid
.
.
/dev/md0: UUID="bfefce58-9128-44a6-94fa-80a9f1d4de27" TYPE="ext4"
.
```

E. Edit `fstab` and add the required entry:

```
UUID="bfefce58-9128-44a6-94fa-80a9f1d4de27"   /disk1
ext4    defaults 0 0
```

F. Mount the filesystems and ensure that they are properly mounted:

```
mount -a
```

G. Verify that NFS and the local filesystem are mounted properly:

```
mount
<serverip>:/mnt/vol1 on /mnt/vol1 type nfs
(rw,bg,clientaddr=<clientip>,vers=4,addr=<serverip>)
/dev/md0 on /disk1 type ext4 (rw)
```

7. Install the SAP compatibility libraries.

Note: This step is required only for an Oracle Linux 6 image. If you are using Oracle Linux 7, skip this step.

Download the `compat-sap-c++-4.8.2-16.el6.x86_64.rpm` file from ULN (linux.oracle.com). The label of the ULN channel is `ol6_x86_64_SAP_addons`.

```
yum -y install compat-sap-c++-4.8.2-16.el6.x86_64.rpm
```

8. Create the target directories for the software installation:

```
mkdir /disk1/usr
mkdir /disk1/usr/sap
mkdir /disk1/oracle
mkdir /disk1/sapmnt
ln -s /disk1/usr/sap /usr/sap
ln -s /disk1/oracle /oracle
ln -s /disk1/sapmnt /sapmnt
```

9. Create a symbolic link to resolve the required shared library:

```
mkdir /usr/sap/lib ; ln -s /opt/rh/SAP/lib64/compat-sap-c++.so
/usr/sap/lib/libstdc++.so.6
```

10. Update the Linux kernel parameters.

A. Edit `/etc/sysctl.conf` and set the parameters as needed. For example:

```
vm.max_map_count = 1000000
kernel.sem = 32000 256000 100 1024
kernel.shmmni = 4096
kernel.shmall = 6294967296
kernel.shmmax = 8398046511104
```

B. Activate the parameters:

```
sysctl -p
```

11. Edit `/etc/security/limits.conf`. For example:

```
sapsys          soft    nofile  38000
sapsys          hard    nofile  38000
sapsys          soft    memlock unlimited
sapsys          hard    memlock unlimited
oracle          soft    nofile  256000
oracle          hard    nofile  256000
oracle          soft    memlock unlimited
oracle          hard    memlock unlimited
root            soft    memlock unlimited
root            hard    memlock unlimited
```

12. Edit `/etc/hosts`. Use either the private or public IP address of the virtual machine compute node depending on your needs. For example:

```
127.0.0.1   localhost localhost.localdomain localhost4
localhost4.localdomain4
::1         localhost localhost.localdomain localhost6
localhost6.localdomain6
<privateip> saphost1.subxxxxxxxx.xxxxxxx.oraclevcn.com saphost1
or
<publicip> saphost1.subxxxxxxxxx.xxxxxxx.oraclevcn.com saphost1
```

13. Reboot.

14. Install the metrics collector for SAP:

```
wget https://objectstorage.eu-frankfurt-
1.oraclecloud.com/p/Dm4E4TfjwaDu7NbcW8-
BQzBtfCWEABo_LCwiNTurCDc/n/imagegen/b/metrics-collector-binary-
store/o/oci-sap-metrics-collector-1.0-8.noarch.rpm
sudo yum -y install oci-sap-metrics-collector-1.0-8.noarch.rpm
service oci-sap-metrics-collector status
oci-sap-metrics-collector is running
```

15. Wait a couple of minutes and run the following command to verify that metric collection works as expected:

```
curl http://127.0.0.1:18181
.
.
  <metric category="config" context="vm" last-update="1528704025" refresh-
interval="60" type="string" unit="none">
    <name>Provider Health Description</name>
    <value>OK</value>
  </metric>
.
.
```

16. Run SWPM until it stops and prompts for installation of Oracle Database software.

SAPinst starts and displays a URL for the browser. Replace the hostname with the IP address of the host that you want to use. Ensure that your security list allows access to the IP address and port displayed. For example:

Provided:

`https://saphost1.subxxxxxxxxxxx.xxxxxx.oraclevcn.com:4237/sapinst/docs/index.html`

Replace with:

`https://aaa.bbb.ccc.ddd:4237/sapinst/docs/index.html`

When SWPM stops and prompts for installation of Oracle Database software, the required `oracle` or `ora<sid>` user (or both) has already been created by SWPM.

```
su - oracle
vncserver -geometry 1280x1024

You will require a password to access your desktops.

Password:
Verify:
xauth:  file /home/oracle/.Xauthority does not exist

New 'saphost1:1 (oracle)' desktop is saphost1:1

Creating default startup script /home/oracle/.vnc/xstartup
Starting applications specified in /home/oracle/.vnc/xstartup
Log file is /home/oracle/.vnc/saphost1:1.log
```

17. Connect via the VNC client, start the software installation, and follow the steps in the Oracle Universal Installer until the software is installed successfully. Note that you have to set up an SSH tunnel for VNC.

    A. Open a console windows (user `oracle`) and install the Oracle Database software. For example:

    ```
    cd /oracle/stage/122/database/SAP
    [oracle@saphost1 ~]$ export DISPLAY=:1
    [oracle@saphost1 ~]$ export DB_SID=MFG
    ./RUNINSTALLER
    ```

    B. Stop the VNC server if it's not required anymore:

    ```
    [oracle@saphost1 SAP]$ vncserver -kill :1
    Killing Xvnc process ID 15368
    ```

18. Install all SAP recommended patches for your release of Oracle Database.

19. Continue with SWPM after Oracle Database software is installed.

20. Update your SAP system with the latest support packages and patches (for example, by running SUM).

Oracle Database in the Cloud

All options and features of Oracle Database 11g Release 2 (11.2.0.4), Oracle Database 12c Release 1 (12.1.0.2), and Oracle Database 12c Release 2 (12.2.0.1), except Oracle Real Application Clusters (RAC) and Oracle Automatic Storage Management (ASM) supported for on-premises deployments of SAP NetWeaver, are supported and certified for Oracle Cloud Infrastructure. SAP customers can therefore use the advanced features Oracle Database In-Memory and Oracle Multitenant in the Oracle Cloud.

Use of Object Storage

Oracle Cloud Infrastructure Object Storage can be consumed as a durable, efficient, and fast destination for backups, and consequently, a restore and recovery source. In contrast to classic file systems, the interface to Object Storage is provided by a SBT_LIBRARY to Recovery Manager (RMAN). Step-by-step instructions are at Backing Up a Database, and at least Java 7 is required to install. This creates the auto-open wallet with the default location of the oracle OS user, `~oracle/hsbtwallet/cwallet.sso`.

The link is established between a Swift Object Store password and the auto-open wallet.

Note: Do *not* run TDE when an auto-open wallet is also present.

This link has the following effects:

- If your cwallet.sso is lost and you can't restore it for any reason, re-create it with the Swift password.
- If you lose your Swift password, get a new one and re-create the cwallet.sso .
- If you lose both your cwallet.sso and your Swift password, create a new Swift password and re-create the wallet.
- You must delete old, unused, and unknown Swift passwords.
- You can back up multiple databases into a bucket.

- You can have multiple buckets configured. Consider changing the configuration file (config_db_name, the `/lib` storage, and the wallet directory). Before you perform any operation, you must adjust RMAN's configuration as follows:

```
CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE'
PARMS  'SBT_LIBRARY=/home/oracle/lib/libopc.so,
SBT_PARMS=(OPC_PFILE=/home/oracle/config2)';
```

Migrating to the Cloud

Various options are available to migrate databases to the cloud. The general approach is described in Migrating Databases to the Cloud in the Oracle Cloud Infrastructure documentation.

This white paper discusses three methods, all of which expect that the source and target platform are running a Linux operating system (Linux x86_64).

RMAN Native via Oracle Cloud Infrastructure Object Storage

On the source host, you configure Object Storage and perform a backup as described in the "Backup and Recovery" subsection later in this section.

On the target host, you configure Object Storage and perform a restore and recovery as described in the "Disaster Recovery Restore" subsection under "Backup and Recovery."

BR*Tools via backup_dev_type=rman_disk

This method is the same as the preceding method, but with BR*Tools integration. See the "Integrating with BR* Tools" subsection under "Backup and Recovery," later in this section. You must perform this step on both the source and target hosts.

1. On the source host, run the following commands:

```
brtools > backup > online_cons
Verification mode: use_rmv
Change 9 (add -pr <password>) (if no TDE is given we need the password
"only" option)
9 - BRBACKUP command line (command) ... [-p initCT3.sap -d rman_disk -t
online_cons -m all -k no -w use_rmv -l E -pr <your_encryption_password>]
```

   Then perform an archive log backup.

   Always change option 9 to include `-pr <your_encryption_password>`.

2. Go to `$SAPDATA_HOME/sapbackup`.

3. In the `back<SID>.log` file, obtain the backup that you intend to use (in this example, `bewbzyys`). *rman_disk* stores the control file in this directory.

```
tar cvf archive.tar bewbzyys bewbzyys.anr backCT3.log
```

4. Copy or move the `archive.tar` file to the target host `$SAPDATA_HOME/sapbackup`.

   As privileged user:

```
chown oracle:oinstall archive.tar ;
```

   As oracle OS user:

```
tar -xvf archive.tar
```

5. On the target host as `<SID>adm`, run the following commands:

```
brtools (5, database reset (4)), for option 9 add -pr
<your_encryption_password>
1 = Select database backup or restore point
```

   For any call to `brrestore` or `brrecover` command lines, copy and edit those lines, and always add `-pr <your_encryption_password>`.

6. Go through the `brrecover` dialogs. You end up with open resetlogs after applying the latest archive log.

BR*Tools via backup_dev_type=stage_copy

This method makes an intermediate backup (for example, to Object Storage) obsolete by using `scp`. The advantage of `scp` is that you don't need extra security list liftings.

For more information, see the “Structure-Retaining Database Copy” section in the SAP Database Guide: Oracle.

Using `scp` requires SSH key pairs between the involved Oracle user accounts on each host.

- The key pairs must be *passwordless*, but not necessarily follow the `id_rsa` or `id_rsa.pub` file names.
- Password protected key pairs won't work. You can verify by using the `scp -B` option.
- Each host’s `~oracle/.ssh/authorized_keys` needs the `.pub` key from the other side.
- Consider removing the SSH key pairs (also from `authorized_keys`) after the job is done.

- When the key file doesn't follow the `id_rsa`/`id_rsa.pub` naming convention, before you call `brbackup / brtools`, run the following command as `<SID>adm`:

```
setenv BR_SCP_CMD "scp -i /path/to/private_key_file"
```

Perform the following steps:

1. On the target host with a preinstalled SAP system, shut down any SAP application server and Oracle Database (as `<SID>adm`).

2. On the source host, edit the `init<SID>.sap` file (or work with an adjusted copy).

```
backup_dev_type=stage_copy
stage_copy_cmd=scp
stage_db_home=/oracle/CT3 (we preserve)
stage_root_dir=/oracle/CT3/sapbackup
archive_stage_dir/oracle/CT3/sapbackup
remote_host=vm1
remote_user=oracle
new_db_home=
```

3. Ensure that the prerequisites in the SAP Database Guide are met (`sapdataX`, `sapbackup`, and `origlogX` directories).

4. Back up to the target host as `<SID>adm`:

```
brbackup -u / -d stage_copy -t online_cons
BR0244I Trying to create remote directory
target:/oracle/<SID>/sapdata2/sr3_1 ...
```

```
brarchive -u / -d stage_copy
```

5. On the target host, run the following command as `<SID>adm`:

```
SQL> startup mount
ORACLE instance started.
Total System Global Area 1946157056 bytes
Fixed Size 2925840 bytes
Variable Size 973081328 bytes
Database Buffers 956301312 bytes
Redo Buffers 13848576 bytes
Database mounted.
SQL> recover database using backup controlfile until cancel;
ORA-00279: change 23321371 generated at 06/28/2017 15:06:52 needed for
thread 1
ORA-00289: suggestion : /oracle/CT3/oraarch/CT3arch1_1790_927893878.dbf
ORA-00280: change 23321371 for thread 1 is in sequence #1790
```

```
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
/oracle/CT3/sapbackup/CT3arch1_1790_927893878.dbf
ORA-00279: change 23322468 generated at 06/28/2017 17:00:32 needed for
thread 1
ORA-00289: suggestion : /oracle/CT3/oraarch/CT3arch1_1791_927893878.dbf
ORA-00280: change 23322468 for thread 1 is in sequence #1791
ORA-00278: log file '/oracle/CT3/sapbackup/CT3arch1_1790_927893878.dbf' no
longer needed for this recovery

Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
cancel
Media recovery canceled.
SQL> alter database open resetlogs;
Database altered.
```

6. On the target host, start the SAP application servers and verify functionality.

7. For security, delete the keys for the user **oracle** between the source and target, unless you have a good reason to keep them.

Backup and Recovery

Decide your backup, restore, and recovery method. The sophisticated methods are Oracle RMAN or SAP BR*Tools, which can integrate with Oracle RMAN. If you use Oracle RMAN, enable Block Change Tracking for backup efficiency.

If you decide to perform backups by using Oracle Cloud Infrastructure Object Storage and have it installed as described earlier, following is how you perform a backup native with Oracle RMAN, encrypted:

```
RMAN> CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE' PARMS
'SBT_LIBRARY=/home/oracle/lib/libopc.so,
SBT_PARMS=(OPC_PFILE=/home/oracle/config)';
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO SBT_TAPE;
RMAN> CONFIGURE BACKUP OPTIMIZATION ON;
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE SBT_TAPE TO '%F';
RMAN> CONFIGURE ENCRYPTION FOR DATABASE ON;
RMAN> CONFIGURE DEVICE TYPE 'SBT_TAPE' PARALLELISM 16;
```

If licensed, also `RMAN> CONFIGURE COMPRESSION ALGORITHM 'MEDIUM';`

It is important to use backup parallelism (16 shown here only as an example) to speed up the process.

Unless you are working with a TDE encrypted database, every RMAN session *requires* the setting of the encryption/decryption password; otherwise, the session fails with a "wallet not open" error.

```
set encryption identified by "yourencryptionpassword" only;
set decryption identified by "yourencryptionpassword";
```

For more information about choosing a backup procedure that meets your needs, see the backup and recovery documentation for your version of Oracle. Be sure to back up regularly to minimize potential data loss and always include a copy of the spfile and the control file.

Creating a Backup

Run the following commands to create a backup:

```
RMAN> connect target /
RMAN> set encryption identified by "yourencryptionpassword" only;
executing command: SET encryption
using target database control file instead of recovery catalog
RMAN> BACKUP INCREMENTAL LEVEL 0 SECTION SIZE 512M DATABASE PLUS ARCHIVELOG;
RMAN> list backup;
```

Validating a Backup

Run the following commands to validate a backup:

```
RMAN> set decryption identified by "yourencryptionpassword";
executing command: SET decryption
using target database control file instead of recovery catalog
RMAN> restore database validate check logical;
```

Integrating with BR* Tools

Consult SAP Notes 113747, 1598594, and 776505.

1. Install the latest BR*Tools patch, as `root`:

   ```
   # mkdir /path/to/backup ; cd /sapmnt/<SID>/exe/uc/linuxx86_64/
   # cp br* /path/to/backup ; rm br*
   # ./SAPCAR -xvf /tmp/DBATL740O11_29-70001657.SAR
   # chown oracle:oinstall brarchive brbackup brconnect brrecover brrestore
   brspace
   # chmod 6774 brarchive brbackup brconnect brrecover brrestore brspace
   # chown <SID>adm:sapsys brtools
   # chmod 755 brtools
   ```

2. Consider write-protecting the files by using `chattr +i`.

3. Currently $SAPDATA_HOME/init<SID>.sap is unchanged, and local backup to disk works out of the box:

```
brbackup -u / -m system -d disk -t online -w use_rmv
```

However, you should change the following parameters in $SAPDATA_HOME/init<SID>.sap:

```
rman_channels = 16
backup_dev_type = rman_disk
rman_sectionsize = 512M
rman_parms = "SBT_LIBRARY=/home/oracle/lib/libopc.so,
SBT_PARMS=(OPC_PFILE=/home/oracle/config)"
```

Without that change, a call to brbackup fails with the following errors:

```
'RMAN-03009: failure of allocate command on sbt_1 channel at 05/17/2017
13:59:25^'
'ORA-19554: error allocating device, device type: SBT_TAPE, device name:
^'
'ORA-27211: Failed to load Media Management Library^'
'Additional information: 2^'
```

Note: In a production environment, you might not want to modify this file. Copy this file and use the copy as the parameter input file (-p profile_file) for BR*Tools. Examples used here refer to the default file for simplicity.

4. Now you can start a full database backup:

```
brbackup -u / -m all -d rman_disk -t online -w use_rmv  -pr
<yourencryptionpassword>
```

An additional control file copy is placed in $SAPDATA_HOME/sapbackup/<tag>, and other brbackup files (log, init<SID>.sap, init<SID>.ora, spfile, and so on) are stored in $SAPDATA_HOME/sapbackup/<SID>.

5. Back up the archived redo logs from disk:

```
brarchive -u / -d rman_disk -w use_rmv -pr <your_encryption_password>
```

6. Create a second backup for archived redo logs that are on disk:

```
brarchive -u / -d rman_disk -w use_rmv -sc -pr <your_encryption_password>
```

7. Back up the database and the archived redo logs all at once without operator interaction:

```
brbackup -u / -m all -d rman_disk -t online -w use_rmv -pr
<your_encryption_password> -c -a -d rman_disk -s -w use_rmv -pr
<your_encryption_password> -c
```

Disaster Recovery Restore

For details, see Recovering a Database from the Object Storage in the Oracle Cloud Infrastructure documentation.

```
curl -u 'your_oracle_cloud_account@domain:#]D>qsYnd<5GCoRM8u0' -v
https://swiftobjectstorage.us-phoenix-1.oraclecloud.com/v1/tenant
```

Get the DBID from control file:

```
curl -u 'your_oracle_cloud_account@domain:#]D>qsYnd<5GCoRM8u0' -v
https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/tenant/bucket?prefix=sbt_catalog/c-
```

For `csh`:

```
curl -u 'your_oracle_cloud_account@domain:#]D>qsYnd<5GCoRM8u0' –v
https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/tenant/bucket\?prefix=sbt_catalog/c-
```

Encryption Considerations

Oracle Transparent Data Encryption (TDE) is recommended for data encryption. It also helps with backup and restore tasks, preventing password-based backups (as people might forget passwords).

For SAP environments, follow SAP Note 974876.

A new installation performed through SWPM creates databases where no tablespace encryption is enabled. To encrypt your SAP data in the database, you must create more tablespaces with encryption enabled and perform a tablespace reorg via BRSPACE.

The following encryption cases are possible:

- **No TDE setup:** This case requires a password for encryption using BR*Tools when using Object Storage-based backups and restore.
- **TDE setup without auto-open wallet:** In this case, using BR*Tools to restore is tedious and needs multiple interactions from a second database instance session running `ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY <encryption_password>`.
- **TDE setup including auto-open wallet:** In this case, using BR*Tools to restore works smoothly and is free from interruptions.

High Availability in the Cloud

This section describes how the Oracle Data Guard toolset can enable high availability for the Oracle Database of an SAP installation running on Oracle Cloud Infrastructure. This document focuses only on physical standby because that is the recommended solution for an SAP environment. The physical standby database runs on a compute instance that needs to fulfil the same SAP system requirements as the primary database, for example, identical operating system user and group IDs. The Oracle Database software needs to be installed using SAP's Software Provisioning Manager (SWPM) to the same location as the primary site (`/oracle/<SID>`) and run on the same release and patch level as the primary database. Ensure that you sufficiently test the reconnect of the SAP instances to the standby database.

Introduction to Oracle Data Guard

Oracle Data Guard ensures high availability, data protection, and disaster recovery for enterprise data. Data Guard provides a comprehensive set of services that create, maintain, manage, and monitor one or more standby databases to enable production Oracle Databases to survive disasters and data corruptions. Data Guard maintains these standby databases as copies of the production database. If the production database becomes unavailable because of a planned or an unplanned outage, Data Guard can switch any standby database to the production role, minimizing the downtime associated with the outage.

Data Guard can be used with traditional backup, restoration, and cluster techniques to provide a high level of data protection and data availability. Data Guard transport services are also used by other Oracle features such as Oracle Streams and Oracle GoldenGate for efficient and reliable transmission of redo data from a source database to one or more remote destinations.

With Data Guard, administrators can optionally improve production database performance by offloading resource-intensive backup and reporting operations to standby systems.

Oracle Data Guard Configurations

An Oracle Data Guard configuration can contain one primary database and up to 30 destinations. The members of a Data Guard configuration are connected by Oracle Net and can be dispersed geographically. Members of a Data Guard configuration can be located anywhere as long as they can communicate with each other. For example, in an Oracle Cloud Infrastructure environment you can have a standby database in the same availability domain as the primary database, along with two standby databases in the same or different availability domains. If you want to make your high-availability setup disaster proof, we recommend having at least one standby database in a different availability domain, preferably in a different region.

You can manage primary and standby databases by using either the SQL command-line interface or the Oracle Data Guard broker interfaces. The broker provides a command-line interface (DGMGRL) and a graphical user interface that is integrated in Oracle Enterprise Manager Cloud Control.

Primary Database

A Data Guard configuration contains one production database, also referred to as the primary database, that functions in the primary role. Your SAP application accesses this database.

Standby Databases

A standby database is a transactional consistent copy of the primary database. Using a backup copy of the primary database, you can create up to 30 standby databases and incorporate them into a Data Guard configuration. After the standby databases are created, Data Guard automatically maintains each one by transmitting redo data from the primary database and then applying the redo to the standby database.

Similar to a primary database, a standby database can be either a single-instance Oracle Database or Oracle RAC.

The types of standby databases are as follows.

Physical Standby Database

A physical standby database provides a physically identical copy of the primary database, with on-disk database structures that are identical to the primary database on a block-for-block basis. The database schema, including indexes, are the same. A physical standby database is kept synchronized with the primary database, through Redo Apply, which recovers the redo data received from the primary database and applies the redo to the physical standby database.

As of Oracle Database 11g Release 1 (11.1), a physical standby database can receive and apply redo data while it is open for read-only access. A physical standby database can therefore be used concurrently for data protection and reporting.

Physical standby is the recommended configuration for an SAP environment.

Logical Standby Database

A logical standby database contains the same logical information as the production database, although the physical organization and structure of the data can be different. The logical standby database is kept synchronized with the primary database through SQL Apply, which transforms

the redo data received from the primary database into SQL statements and then executes the SQL statements on the standby database.

The flexibility of a logical standby database lets you upgrade Oracle Database software (patch sets and new Oracle Database releases) and perform other database maintenance in rolling fashion with almost no downtime. In Oracle Database 11g and later, the transient logical-database rolling-upgrade process can also be used with existing physical standby databases.

Note: Use logical standby databases only for certain operations, such as migrations. Do not use them for SAP production environments.

Snapshot Standby Database

A snapshot standby database is a fully updatable standby database. Like a physical or logical standby database, a snapshot standby database receives and archives redo data from a primary database. Unlike a physical or logical standby database, a snapshot standby database does not apply the redo data that it receives. The redo data is applied only when the snapshot standby database is converted back into a physical standby database, after first discarding any local updates made to the snapshot standby database.

A snapshot standby database is best used in scenarios that require a temporary, updatable snapshot of a physical standby database. For example, you can use the Oracle Real Application Testing option to capture a primary database workload and then replay it for test purposes on the snapshot standby. Note that because redo data received by a snapshot standby database is not applied until it is converted back into a physical standby, the time needed to recover from a primary database failure is directly proportional to the amount of redo data that needs to be applied.

Configuration Example

The following figure shows a typical Oracle Data Guard configuration that contains a primary database that transmits redo data to a standby database. The standby database is remotely located from the primary database for disaster recovery and backup operations.

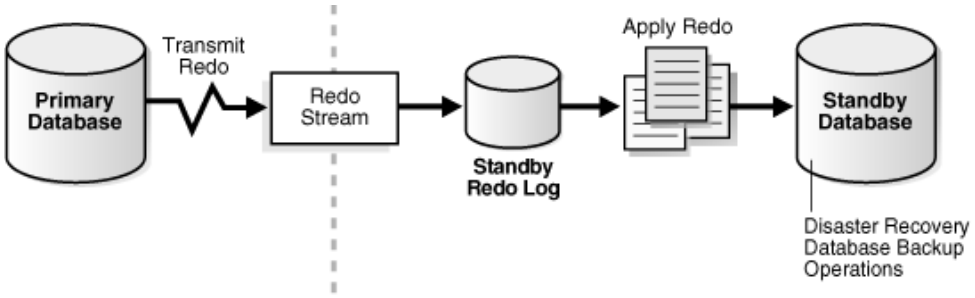

Figure 2: Typical Oracle Data Guard Configuration

Oracle Data Guard Services

This section explains how Oracle Data Guard manages the transmission of redo data, the application of redo data, and changes to the database roles.

Redo Transport Services

Redo transport services control the automated transfer of redo data from the production database to one or more archival destinations. Redo transport services perform the following tasks:

- Transmit redo data from the primary system to the standby systems in the configuration.
- Manage the process of resolving any gaps in the archived redo log files caused by a network failure.
- Automatically detect missing or corrupted archived redo log files on a standby system and automatically retrieve replacement archived redo log files from the primary database or another standby database.

For more information, see Redo Transport Services.

Apply Services

The redo data transmitted from the primary database is written to the standby redo log on the standby database. Apply services automatically apply the redo data on the standby database to maintain consistency with the primary database. They also allow read-only access to the data.

The main difference between physical and logical standby databases is the manner in which apply services apply the archived redo data.

For physical standby databases, Data Guard uses Redo Apply technology, which applies redo data on the standby database by using standard recovery techniques of an Oracle Database, as shown in following figure:

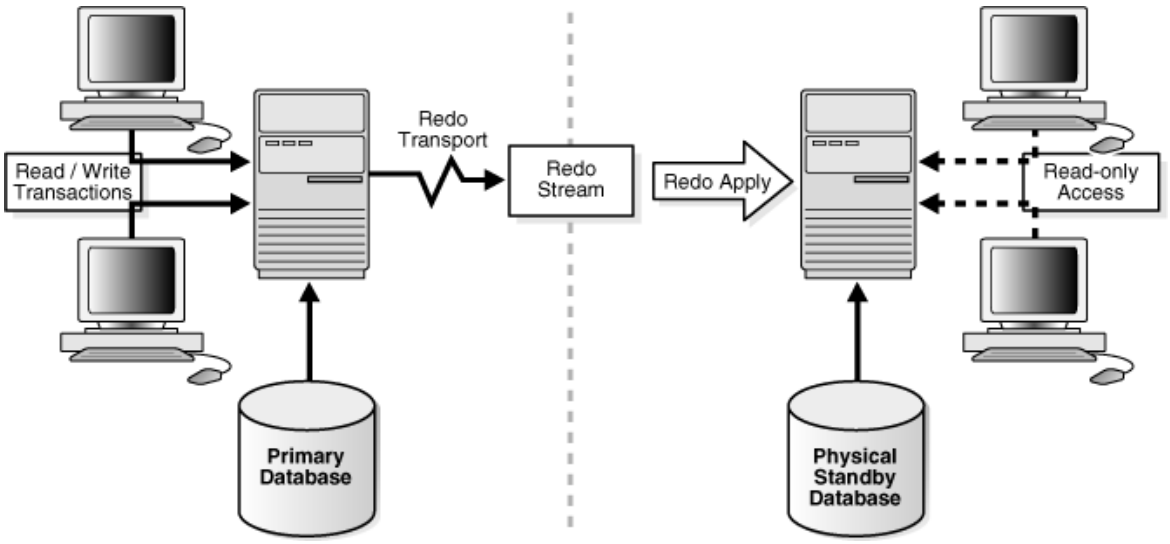

Figure 3: Automatic Updating of a Physical Standby Database

For more information, see Apply Services.

Role Transitions

An Oracle Database operates in one of two roles: primary or standby. Using Data Guard, you can change the role of a database by using either a switchover operation or a failover operation.

- A switchover is a role reversal between the primary database and one of its standby databases. A switchover ensures no data loss. This is typically done for planned maintenance of the primary system. During a switchover, the primary database transitions to a standby role, and the standby database transitions to the primary role.
- A failover occurs when the primary database is unavailable. Failover is performed only if the primary database fails, and the failover results in a transition of a standby database to the primary role. The database administrator can configure Data Guard to ensure no data loss.

The role transitions described in this white paper are invoked manually by using SQL statements. You can also use the Data Guard broker to simplify role transitions and automate failovers through the Oracle Enterprise Manager Cloud Control GUI or the DGMGRL command-line interface, as described in Section 1.3 of the *Data Guard Concepts and Administration* guide. For more information, see Role Transitions.

Oracle Data Guard Broker

The Oracle Data Guard broker is a distributed management framework that automates the creation, maintenance, and monitoring of Oracle Data Guard configurations. You can use either the Oracle Enterprise Manager Cloud Control GUI or the Data Guard command-line interface (DGMGRL) to perform the following tasks:

- Create and enable Data Guard configurations, including setting up redo transport services and apply services.
- Manage an entire Data Guard configuration from any system in the configuration.
- Manage and monitor Data Guard configurations that contain Oracle RAC primary or standby databases.
- Simplify switchover and failover by allowing you to invoke them with either a single key click in Oracle Enterprise Manager Cloud Control or a single command in the DGMGRL command-line interface.
- Enable Data Guard fast-start failover to fail over automatically when the primary database becomes unavailable. When fast-start failover is enabled, the Data Guard broker determines if a failover is necessary and initiates the failover to the specified target standby database automatically, with no need for DBA intervention.

In addition, Oracle Enterprise Manager Cloud Control automates and simplifies the following tasks:

- Creating a physical or logical standby database from a backup copy of the primary database
- Adding new or existing standby databases to an existing Data Guard configuration
- Monitoring log apply rates, capturing diagnostic information, and detecting problems quickly with centralized monitoring, testing, and performance tools

The DGMGRL command-line interface allows you to control and monitor a Data Guard configuration from the DGMGRL prompt or within scripts. You can perform most of the activities required to manage and monitor the databases in the configuration by using DGMGRL. For complete DGMGRL reference information and examples, see Oracle Data Guard Broker. This white paper focuses on using this command-line interface.

Oracle Data Guard Protection Modes

In some situations, a business cannot afford to lose data regardless of the circumstances. In other situations, the availability of the database might be more important than any potential data loss in the unlikely event of multiple failures. Finally, some applications require maximum database performance at all times, and can therefore tolerate a small amount of data loss if any component should fail. The following sections summarize the three distinct modes of data protection.

All three protection modes require that specific redo transport options be used to send redo data to at least one standby database. For more information about setting the protection mode of a primary database, see Oracle Data Guard Protection Modes.

Maximum Availability

This protection mode provides the highest level of data protection that is possible without compromising the availability of a primary database. With Oracle Data Guard, transactions do not commit until all redo data required to recover those transactions has either been received in memory or written to the standby redo log (depending upon configuration) on at least one synchronized standby database. If the primary database cannot write its redo stream to at least one synchronized standby database, it operates as if it were in maximum performance mode. This preserves primary database availability until it is able to write its redo stream to a synchronized standby database again.

This protection mode ensures zero data loss except in the case of certain double faults, such as failure of a primary database after failure of the standby database.

Maximum Performance

Maximum performance is the default protection mode. It provides the highest level of data protection that is possible without affecting the performance of a primary database. This is accomplished by allowing transactions to commit when all redo data generated by those transactions has been written to the online log. Redo data is also written to one or more standby databases. However, this is done asynchronously for transaction commitment, so primary database performance is unaffected by delays in writing redo data to the standby databases.

This protection mode offers slightly less data protection than maximum availability mode and has minimal impact on primary database performance.

Maximum Protection

This protection mode ensures that no data loss occurs if the primary database fails. To provide this level of protection, the redo data required to recover a transaction must be written to both the online redo log and to the standby redo log on at least one synchronized standby database before the transaction commits. To ensure that data loss cannot occur, the primary database shuts down, rather than continuing to process transactions, if it cannot write its redo stream to at least one synchronized standby database.

Client Failover

A high-availability architecture requires a fast failover capability for databases and database clients. Client failover encompasses failure notification, stale-connection cleanup, and transparent reconnection to the new primary database. Oracle Database provides the capability to integrate database failover with failover procedures that automatically redirect clients to a new primary database within seconds of a database failover.

Oracle Data Guard and Complementary Technologies

Oracle Database provides several unique technologies that complement Oracle Data Guard to help keep business-critical systems running with greater levels of availability and data protection than when using any one solution by itself. The following list summarizes some Oracle high-availability technologies.

Flashback Database

The Flashback Database feature provides fast recovery from logical data corruption and user errors. By allowing you to flash back in time, previous versions of business information that might have been erroneously changed or deleted can be accessed once again. This feature provides the following benefits:

- Eliminates the need to restore a backup and roll forward changes up to the time of the error or corruption. Instead, Flashback Database can roll back an Oracle Database to a previous point-in-time, without restoring data files.
- Provides an alternative to delaying the application of redo data to protect against user errors or logical corruptions. Therefore, standby databases can be more closely synchronized with the primary database, thus reducing failover and switchover times.
- Avoids the need to completely re-create the original primary database after a failover. The failed primary database can be flashed back to a point in time before the failover and converted to be a standby database for the new primary database.

For information about Flashback Database, see the Oracle Database Backup and Recovery User's Guide. For information about the application of redo data, see Section 8.2.2 of the *Data Guard Concepts and Administration* guide.

Recovery Manager (RMAN)

RMAN is an Oracle utility that simplifies backing up, restoring, and recovering database files. Like Oracle Data Guard, RMAN is a feature of the Oracle Database and does not require separate installation. Data Guard is well integrated with RMAN, allowing you to perform the following tasks:

- Use the Recovery Manager `DUPLICATE` command to create a standby database from backups of your primary database.
- Take backups on a physical standby database instead of the production database, relieving the load on the production database and enabling efficient use of system resources on the standby site. Also, backups can be taken while the physical standby database is applying redo data.
- Help manage archived redo log files by automatically deleting the archived redo log files used for input after performing a backup.

For more information, see Creating a Standby Database with Recovery Manager in the *Data Guard Concepts and Administration* guide.

Summary of Oracle Data Guard Benefits

Oracle Data Guard provides these benefits:

- **Disaster recovery, data protection, and high availability**: Data Guard provides an efficient and comprehensive solution for disaster recovery and high availability. Easy-to-manage switchover and failover capabilities allow role reversals between primary and standby databases, minimizing the downtime of the primary database for planned and unplanned outages.
- **Complete data protection**: Oracle Data Guard can ensure zero data loss, even in the face of unforeseen disasters. A standby database provides a safeguard against unplanned outages of all types, including data corruption and administrative error. Because the redo data received from a primary database is validated at a standby database, physical corruptions that can occur at a primary database are not propagated to the standby database. Additional validation performed at a standby database also prevents logical intra-block corruptions and lost-write corruptions from propagating to the standby. Also, administrative errors such as accidental file deletions by a storage administrator are not propagated to a standby database. A physical standby database can also be used to protect against user errors either by delaying the redo apply or by using Flashback Database to rewind the standby and extract a good copy of the data.
- **Efficient use of system resources**: The standby database tables that are updated with redo data received from the primary database can be used for other tasks such as backups, reporting, summations, and queries. This reduces the primary database workload necessary to perform these tasks, saving valuable CPU and I/O cycles.

- **Flexibility in data protection to balance availability against performance requirements**: Data Guard offers maximum protection, maximum availability, and maximum performance modes to help enterprises balance data availability against system performance requirements.
- **Automatic gap detection and resolution**: If connectivity is lost between the primary and one or more standby databases (for example, because of network problems), redo data that is generated on the primary database cannot be sent to those standby databases. When a connection is reestablished, the missing archived redo log files (referred to as a gap) are automatically detected by Data Guard, which then automatically transmits the missing archived redo log files to the standby databases. The standby databases are synchronized with the primary database, without manual intervention by the DBA.
- **Centralized and simple management**: The Data Guard broker provides a graphical user interface and a command-line interface to automate management and operational tasks across multiple databases in a Data Guard configuration. The broker also monitors all the systems within a single Data Guard configuration.
- **Integration with Oracle Database**: Data Guard is a feature of Oracle Database Enterprise Edition and does not require separate installation.
- **Automatic role transitions**: When fast-start failover is enabled, the Data Guard broker automatically fails over to a synchronized standby site in the event of a disaster at the primary site. This action requires no intervention by the DBA. In addition, applications are automatically notified of the role transition.

References

SAP

Most links need SAP login credentials.

SAP Documentation

- SAP Product Availability Matrix (PAM)
- SAP Software Logistics Toolset (SL Tools)
- SAP Download Manager
- SAP Software Download Center (SWDC)
- SAP Guide Finder
- SAP Community Network - Oracle Community
- SAP Help TCP/IP Ports of All SAP Products

SAP Notes

- 2474949 - SAP NetWeaver® on Oracle Cloud Infrastructure
- 2520061 - SAP on Oracle Cloud Infrastructure: Support prerequisites
- 2655715 – SAP on Linux with Oracle Cloud Infrastructure Compute: Enhanced Monitoring
- 1565179 – SAP software and Oracle Linux
- 1635808 – Oracle Linux 6.x: SAP-Installation and Upgrade
- 2069760 - Oracle Linux 7.x SAP Installation and Upgrade
- 1597355 - Swap-space recommendation for Linux
- 1770532 - HugePages on Linux for Oracle Database
- 1672954 - Oracle 11g and 12c: Usage of hugepages on Linux
- 1871318 - Linux: Disable Transparent HugePages for Oracle Database
- 1114181 - Oracle Database 11g - file system support on Linux
- 2171857 - Oracle Database 12c - file system support on Linux
- 146505 - SAP GUI for the Java Environment
- 1431800 - Oracle 11.2.0: Central Technical Note
- 1914631 - Central Technical Note for Oracle Database 12c Release 1 (12.1)
- 2470660 - Oracle Database Central Technical Note for 12c Release 2 (12.2)
- 1868094 - Overview: Oracle Security SAP Notes
- 974876 - Oracle Transparent Data Encryption (TDE)
- 1598594 - BR*Tools configuration for Oracle installation using user "oracle"
- 113747 - Owners and authorizations of BR*Tools
- 776505 - ORA-01017/ORA-01031 in BR*Tools on Linux and Solaris 11

Oracle

- Oracle Cloud Infrastructure
- Oracle Cloud Hosting and Delivery Policies
- Oracle Database
- Oracle Linux
- Oracle-SAP Solutions site

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

blogs.oracle.com/oracle
facebook.com/oracle
twitter.com/oracle
oracle.com/SAP

Integrated Cloud Applications & Platform Services

SAP NetWeaver® Application Server ABAP/Java on Oracle Cloud Infrastructure
September 2018
Author: Oracle Corporation
Contributing Authors: Jan Boonen, Torsten Grambs, Jan Klokkers, Christoph Kurucz, Thomas Schuele, Eder Zechim, Markus Breunig, Gilson Melo