General SAP Guides

SAP Guides
# Table of Contents

*Home* .................................................................................................................... 1  
*SAP on AWS Overview and Planning* .......................................................................................... 2  
  
  *AWS overview* .................................................................................................................. 2  
  *AWS services* .................................................................................................................... 2  
  *AWS global infrastructure* .................................................................................................. 5  
  *AWS security and compliance* ............................................................................................ 5  
  *AWS resource provisioning and management* ....................................................................... 6  
*SAP on AWS overview* ............................................................................................................. 6  
  *SAP software and licenses on AWS* ..................................................................................... 6  
  *SAP support on AWS* ......................................................................................................... 7  
  *Deploying SAP systems on AWS* ........................................................................................ 8  
  *Partner services* ................................................................................................................. 9  
*SAP on AWS planning* ............................................................................................................. 10  
  *SAP notes* ........................................................................................................................ 10  
  *SAP on AWS architectures* ................................................................................................. 10  
  *Choosing an AWS Region and Availability Zone* ................................................................ 12  
  *Network and connectivity* .................................................................................................. 12  
  *Following security best practices* ....................................................................................... 5  
  *EC2; instance types for SAP* ............................................................................................... 14  
  *Operating systems* .......................................................................................................... 15  
  *Databases* ........................................................................................................................ 17  
  *SAP installation media* ....................................................................................................... 18  
  *SAProuter and SAP Solution Manager* ............................................................................... 18  
*Additional reading* ................................................................................................................. 19  
*Document revisions* ................................................................................................................ 19  
*AWS Data Provider for SAP* .................................................................................................. 20  
  
  *Introduction* ...................................................................................................................... 20  
  *Upgrading from earlier versions* ......................................................................................... 21  
  *Technical requirements* .................................................................................................... 21  
  *Amazon VPC Endpoints* ................................................................................................... 22  
  *IAM Roles* ......................................................................................................................... 24  
*Installing the Data Provider* ................................................................................................... 27  
  *Installing on Linux* .......................................................................................................... 28  
  *Installing on Red Hat and Oracle Enterprise Linux* .......................................................... 31  
  *Installing on Windows* ...................................................................................................... 32  
*Updating the Data Provider* ................................................................................................... 38  
  *Uninstalling on Linux* ....................................................................................................... 38  
  *Uninstalling on Windows* .................................................................................................. 38  
*Troubleshooting* .................................................................................................................... 39  
  *Troubleshooting on Linux* ............................................................................................... 39  
  *Troubleshooting on Windows* .......................................................................................... 42  
*Appendix A: Customizing the AWS Data Provider* ............................................................ 46  
  *Syntax Rules for configuration files* .................................................................................. 46  
  *User-configurable EC2 instance types* .............................................................................. 47  
  *User-configurable EBS volume types* ................................................................................. 48  
  *User-configurable support status* .................................................................................... 49  
*Appendix B: Verification of AWS Data Provider for SAP in SAP system monitoring* ............. 49  
  *Checking Metrics with SAPOSCOL* .................................................................................. 49  
  *Checking Metrics with SAP CCMS* ............................................................................... 51  
*Appendix C: Changes and bug fixes* .................................................................................... 54  
*Appendix D: Uninstalling older 2.X versions* ......................................................................... 56  
*Appendix E: Example of Captured Metrics* ......................................................................... 56
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage</td>
<td>114</td>
</tr>
<tr>
<td>Network</td>
<td>114</td>
</tr>
<tr>
<td>Business Continuity</td>
<td>114</td>
</tr>
<tr>
<td>High Availability</td>
<td>114</td>
</tr>
<tr>
<td>Deployment</td>
<td>115</td>
</tr>
<tr>
<td>Step 1: Db2 Virtual Hostname</td>
<td>116</td>
</tr>
<tr>
<td>Step 2: AWS Overlay IP</td>
<td>116</td>
</tr>
<tr>
<td>Step 3: AWS Resources</td>
<td>116</td>
</tr>
<tr>
<td>Step 4: SAP Netweaver and IBM Db2 Deployment</td>
<td>116</td>
</tr>
<tr>
<td>Step 5: Db2 HADR Setup</td>
<td>120</td>
</tr>
<tr>
<td>Step 6: Pacemaker Cluster Setup</td>
<td>123</td>
</tr>
<tr>
<td>Step 6a: Setup on RHEL</td>
<td>123</td>
</tr>
<tr>
<td>Step 6b: Setup on SLES</td>
<td>128</td>
</tr>
<tr>
<td>Step 7: Post Setup Configuration</td>
<td>135</td>
</tr>
<tr>
<td>Step 8: Testing and Validation</td>
<td>137</td>
</tr>
<tr>
<td>Operations</td>
<td>138</td>
</tr>
<tr>
<td>Monitoring</td>
<td>138</td>
</tr>
<tr>
<td>Backup and Recovery</td>
<td>138</td>
</tr>
<tr>
<td>Operating System Maintenance</td>
<td>140</td>
</tr>
<tr>
<td>Appendix 1: Testing on RHEL Setup</td>
<td>141</td>
</tr>
<tr>
<td>Test Case 1: Manual Failover</td>
<td>141</td>
</tr>
<tr>
<td>Test Case 2: Shut Down the Primary EC2 Instance</td>
<td>143</td>
</tr>
<tr>
<td>Test Case 3: Stop the Db2 Instance on the Primary Instance</td>
<td>144</td>
</tr>
<tr>
<td>Test Case 4: End the Db2 Process (db2sysc) on the Node that Runs the Primary Database</td>
<td>145</td>
</tr>
<tr>
<td>Test Case 5: End the Db2 Process (db2sysc) on the Node that Runs the Standby Database</td>
<td>145</td>
</tr>
<tr>
<td>Test Case 6: Simulating a Crash of the Node that Runs the Primary Db2</td>
<td>146</td>
</tr>
<tr>
<td>Appendix 2: Testing on SLES Setup</td>
<td>147</td>
</tr>
<tr>
<td>Test Case 1: Manual Failover</td>
<td>147</td>
</tr>
<tr>
<td>Test Case 2: Shut Down the Primary EC2 Instance</td>
<td>149</td>
</tr>
<tr>
<td>Test Case 3: Stop the Db2 Instance on the Primary Instance</td>
<td>149</td>
</tr>
<tr>
<td>Test Case 4: End the Db2 Process (db2sysc) on the Node that Runs the Primary Database</td>
<td>150</td>
</tr>
<tr>
<td>Test Case 5: End the Db2 Process (db2sysc) on the Node that Runs the Standby Database</td>
<td>151</td>
</tr>
<tr>
<td>Test Case 6: Simulating a Crash of the Node that Runs the Primary Db2</td>
<td>152</td>
</tr>
<tr>
<td>FAQ</td>
<td>153</td>
</tr>
<tr>
<td>Contributors</td>
<td>153</td>
</tr>
<tr>
<td>Additional Reading</td>
<td>153</td>
</tr>
<tr>
<td>Document Revisions</td>
<td>153</td>
</tr>
<tr>
<td>Notices</td>
<td>154</td>
</tr>
<tr>
<td>Notices</td>
<td>155</td>
</tr>
</tbody>
</table>
General SAP Guides

This section of the SAP on AWS technical documentation provides overviews and planning information for SAP users and partners, including general information about implementing, configuring, and operating SAP environments and solutions on the AWS Cloud. This section includes the following guides:

- SAP on AWS Overview and Planning (p. 2)
- AWS Data Provider for SAP (p. 20)
- SAP on AWS Pricing and Optimization (p. 61)
- Architecture Guidance for Availability and Reliability of SAP on AWS (p. 79)
- SAP IBM Pacemaker (p. 110)

For information about specific SAP products, see the following sections of this documentation set:

- SAP HANA technical guides
- SAP NetWeaver technical guides
- SAP BusinessObjects technical guides

About this content set

SAP on AWS technical documentation provides detailed information on how to migrate, implement, configure, and operate SAP solutions on AWS.

Additional resources from AWS

- SAP and AWS: announcements, solutions, support, pricing, FAQ
- Find an AWS SAP partner
- AWS for SAP blog
- Case Studies
- AWS presentations from SAPPHIRE NOW 2018
- Questions and support

Resources from SAP

- SAP notes and Knowledge Base articles
- SAP Note 1656250: SAP on AWS: Supported instance types (requires SAP One Support Launchpad user account)
SAP on AWS Overview and Planning

*SAP specialists, Amazon Web Services (AWS)*

*Last updated (p. 19): May 2019*

This guide provides overview and planning information for SAP customers and partners who are considering implementing or migrating SAP environments or systems to the Amazon Web Services (AWS) Cloud.

This guide is intended for users who have previous experience installing, migrating, and operating SAP environments and systems on traditional on-premises infrastructure. It consists of three main sections:

- An overview of the AWS Cloud and AWS services (p. 2), for readers who are new to the cloud.
- An overview of SAP on AWS (p. 6), including software and licenses, support options, and partner services.
- Technical considerations (p. 10) that will help you plan and get the most out of your SAP environment on AWS.

**Note**
To access the SAP notes referenced in this guide, you must have an SAP One Support Launchpad user account. For more information, see the SAP Support website.

### About this Guide

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the AWS Cloud. For the other guides in the series, ranging from overviews to advanced topics, see https://aws.amazon.com/sap/docs/.

### AWS Overview

AWS offers a broad set of global, cloud-based services, including compute, storage, networking, Internet of Things (IoT), and many others. These services help organizations move faster, lower IT costs, and support scalability. AWS is trusted by the largest enterprises and popular start-ups to power a wide variety of workloads, such as web and mobile applications, game development, data processing and warehousing, storage, and archiving.

### AWS Services

AWS provides over 165 cloud services that you can use in combinations tailored to your business or organizational needs. For information about all AWS services, see the Amazon Web Services Cloud Platform whitepaper.

This section introduces the AWS services that are most relevant for the deployment and operation of SAP solutions. The following list provides a high-level description of each service and its use for SAP systems. To view features, pricing, and documentation for an individual service, follow the details link after the description.
<table>
<thead>
<tr>
<th>Area</th>
<th>Service</th>
<th>Description</th>
<th>SAP uses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compute</td>
<td>Amazon Elastic Compute Cloud (Amazon EC2)</td>
<td>Secure, resizable compute capacity in the cloud. (<a href="#">details</a>)</td>
<td>Virtual and bare metal servers for the installation and operation of SAP systems.</td>
</tr>
<tr>
<td>Storage</td>
<td>Amazon Elastic Block Store (Amazon EBS)</td>
<td>Persistent block storage volumes for use with EC2 instances. (<a href="#">details</a>)</td>
<td>File systems for SAP software (e.g., <code>/usr/sap</code>), SAP database log and data files, and SAP local backups.</td>
</tr>
<tr>
<td></td>
<td>Amazon Simple Storage Service (Amazon S3)</td>
<td>Object storage service that offers an extremely durable, highly available, and infinitely scalable data storage infrastructure. (<a href="#">details</a>)</td>
<td>Highly durable storage for file backups, database backups, archiving data, data lakes, and more.</td>
</tr>
<tr>
<td></td>
<td>Amazon Elastic File System (Amazon EFS)</td>
<td>Simple, scalable, elastic file system for Linux-based workloads for use with AWS Cloud services and on-premises resources. (<a href="#">details</a>)</td>
<td>Shared file system for SAP application servers (e.g., <code>/sapmnt</code>).</td>
</tr>
<tr>
<td></td>
<td>Amazon FSx for Windows File Server</td>
<td>Fully managed, highly durable, and available native Microsoft Windows file system. (<a href="#">details</a>)</td>
<td>Shared file system for SAP application servers (e.g., <code>/sapmnt</code>).</td>
</tr>
<tr>
<td>Networking</td>
<td>Amazon Virtual Private Cloud (Amazon VPC)</td>
<td>Logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. (<a href="#">details</a>)</td>
<td>Network for SAP resources. You can control the level of isolation of your EC2 instance from other networks, instances, and on-premises network resources, such as those in production and non-production environments.</td>
</tr>
<tr>
<td></td>
<td>AWS Site-to-Site VPN</td>
<td>Enables you to securely connect your on-premises network or branch office site to your VPC. (<a href="#">details</a>)</td>
<td>Network connectivity between on-premises systems/users and SAP systems on AWS.</td>
</tr>
<tr>
<td></td>
<td>AWS Direct Connect</td>
<td>Lets you establish private network connectivity between AWS and your data center, office, or co-</td>
<td>Private network connectivity between on-premises systems/users and the SAP</td>
</tr>
<tr>
<td>Area</td>
<td>Service</td>
<td>Description</td>
<td>SAP uses</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Area</td>
<td>Location environment. (details)</td>
<td>system or environment on AWS.</td>
<td></td>
</tr>
<tr>
<td>Area</td>
<td>Amazon Route 53</td>
<td>Highly available and scalable cloud Domain Name System (DNS) web service.</td>
<td>Name and address resolution for SAP systems running on AWS.</td>
</tr>
<tr>
<td>Area</td>
<td>Amazon Time Sync</td>
<td>Highly accurate and reliable time reference that is natively accessible from EC2 instances. (Linux</td>
<td>Time synchronization for your SAP systems on EC2 instances.</td>
</tr>
<tr>
<td>Management tools</td>
<td>AWS Management Console</td>
<td>Simple web interface to provision and manage AWS resources. (details)</td>
<td>Provisioning and management of AWS resources for your SAP environment on AWS.</td>
</tr>
<tr>
<td>Management tools</td>
<td>AWS Command Line Interface (AWS CLI)</td>
<td>Command-line tool set to provision and manage AWS resources. (details)</td>
<td>Creation of scripts to automate the provisioning and management of AWS resources for your SAP environment on AWS.</td>
</tr>
<tr>
<td>Management tools</td>
<td>AWS CloudFormation</td>
<td>An easy way to create a collection of related AWS resources and provision them in an orderly and predictable fashion. (details)</td>
<td>Automated provisioning of AWS resources for new SAP landscapes, disaster recovery environments, and other use cases.</td>
</tr>
<tr>
<td>Management tools</td>
<td>Amazon CloudWatch</td>
<td>Monitoring for AWS Cloud resources and the applications you run on AWS: collect and track metrics, collect and monitor log files, and set alarms. (details)</td>
<td>Monitoring SAP systems running on AWS.</td>
</tr>
<tr>
<td>Management tools</td>
<td>AWS CloudTrail</td>
<td>Records activity made on your account and delivers log files to your S3 bucket. (details)</td>
<td>Audit capabilities within your AWS account, such as use of the Amazon EC2 API.</td>
</tr>
<tr>
<td>Area</td>
<td>Service</td>
<td>Description</td>
<td>SAP uses</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>----------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Security, identity, and compliance</td>
<td>AWS Identity and Access Management (IAM)</td>
<td>Manages access to AWS services and resources securely. Using IAM, you can</td>
<td>Fine-grained access control using a least privileged security model to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>create and manage AWS users and groups, and use permissions to allow and</td>
<td>access specific AWS services and actions; e.g., to allow SAP BASIS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>deny their access to AWS resources. (details)</td>
<td>resources to launch, to stop and start EC2 instances without terminating</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>them.</td>
</tr>
</tbody>
</table>

**AWS Global Infrastructure**

The AWS Cloud infrastructure is built around Regions and Availability Zones. An AWS Region is a physical location that provides multiple, physically separated and isolated Availability Zones. Each Availability Zone consists of one or more data centers that are connected with low-latency, high-throughput, and highly redundant networking. These Availability Zones offer an easier and more effective way to design and operate your applications and databases, making them more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For a list of the available AWS Regions and to learn more about the AWS global infrastructure, see [Global Infrastructure](https://aws.amazon.com/global-infrastructure/) on the AWS website.

**AWS Security and Compliance**

**Security**

At AWS, security is our top priority. As an AWS customer, you will benefit from a data center and network architecture built to meet the requirements of the most security-sensitive organizations. Security in the cloud is much like security in your on-premises data centers—only without the costs of maintaining facilities and hardware. In the cloud, you don't have to manage physical servers or storage devices. Instead, you use software-based security tools to monitor and protect the flow of information into and out of your cloud resources.

As an AWS customer you inherit all the best practices of AWS policies, architecture, and operational processes built to satisfy the requirements of our most security-sensitive customers, and get the flexibility and agility you need in security controls.

The AWS Cloud enables a shared responsibility model. While AWS manages security of the cloud, you are responsible for security in the cloud. This means that you retain control of the security you choose to implement to protect your own data, platform, applications, systems, and networks no differently than you would in an on-site data center.

To learn more about AWS security, see [AWS Cloud Security](https://aws.amazon.com/cloud-security/) on the AWS website.

**Compliance**

AWS provides robust controls to help maintain security and data protection in the cloud. As systems are built on top of AWS Cloud infrastructure, compliance responsibilities will be shared. By tying together governance-focused, audit-friendly service features with applicable compliance or audit standards, AWS compliance enablers build on traditional programs and help you operate in an AWS security control environment.
The IT infrastructure that AWS provides to its customers is designed and managed in alignment with best security practices and a variety of IT security standards. The following is a partial list of assurance programs with which AWS complies:

- SOC 1/ISAE 3402, SOC 2, SOC 3
- FISMA, DIACAP, and FedRAMP
- PCI DSS Level 1
- ISO 9001, ISO 27001, ISO 27018

To learn more about AWS compliance, see AWS Cloud Compliance on the AWS website.

**AWS Provisioning and Management**

The provisioning and management of AWS services and resources use a self-service model managed by the customer or a partner. For an overview of the tools available for provisioning and management, see the management tools in the AWS Services (p. 2) section.

Figure 1 shows the services managed by AWS and the services managed by the customer or partner for SAP.

![Figure 1: Managed services for SAP on AWS](image)

**SAP on AWS Overview**

AWS has been working with SAP since 2011 to help customers deploy and migrate their SAP applications to AWS, and SAP supports running the vast majority of available SAP applications on AWS. In addition, AWS is working with SAP to power multiple SaaS services and offerings, such as SAP Concur, SAP SuccessFactors, SAP Cloud Platform, and SAP HANA Enterprise Cloud.

**SAP Software and Licenses on AWS**

This section describes the options available for SAP software and licenses on AWS.
Bring Your Own Software and License

The majority of SAP solutions that can be run on AWS use a bring-your-own-software and bring-your-own-license (BYOL) model. Running SAP systems on AWS doesn't require special or new SAP licenses. If you’re an existing SAP customer, you can use your existing SAP licenses when running SAP on AWS. You are responsible for obtaining a valid SAP license, and you must ensure that you are in compliance with the SAP licensing policies. AWS does not provide or sell SAP licenses.

AWS Marketplace

AWS Marketplace is a digital catalog with thousands of software listings from independent software vendors that makes it easy to find, test, buy, and deploy software that runs on AWS. To view SAP-related offerings available in AWS Marketplace, follow this link: SAP in AWS Marketplace.

SAP Trial and Developer Licenses

The SAP Cloud Appliance Library provides access to an online repository of the latest preconfigured SAP solutions. You can quickly deploy these solutions on AWS by using a launch wizard that automates deployment. Some of the solutions available in the SAP Cloud Appliance Library are provided with free trial or developer edition licenses.

SAP Hardware Key Generation

SAP hardware key generation on EC2 instances uses a specific process that is dependent on the SAP kernel patch level. If a hardware key is generated before patching the SAP kernel to the proper level, and the kernel is updated at a later time, the hardware key may change, making the installed license invalid. For details on how the SAP hardware ID is generated on EC2 instances and the required SAP kernel patch levels see the following SAP notes (SAP One Support Launchpad access required):

- SAP Note 1178686 – Linux: Alternative method to generate a SAP hardware key
- SAP Note 2327159 – SAP NW License Behavior in Virtual and Cloud Environments
- SAP Note 1697114 – Determination of hardware ID In Amazon clouds
- SAP Note 2113263 – Additional public key for AWS Hardware ID
- SAP Note 2319387 – Adjustment of the license check for AWS China

SAP Support on AWS

AWS and SAP have worked together closely to ensure that you receive the same level of support via the same support channels, whether you’re running your SAP systems on AWS or on premises.

SAP Solutions Supported on AWS

The majority of SAP solutions that run on traditional on-premises infrastructure are fully supported by SAP on AWS. For the complete list of SAP solutions supported on AWS, see SAP Note 1656099 and the other notes referenced within that note.

SAP Support on AWS

To ensure full support of your SAP on AWS environment from SAP and AWS, you must follow the guidelines and requirements in SAP Note 1656250. Here are the primary requirements you must follow to ensure support of your SAP on AWS environment:
• Enable detailed monitoring for Amazon CloudWatch on each EC2 instance to ensure that the required AWS metrics are provided in one-minute intervals. For additional information on Amazon CloudWatch, see https://aws.amazon.com/cloudwatch.

• Install, configure, and run the AWS Data Provider for SAP (p. 20) on each EC2 instance. The AWS Data Provider collects the required performance and configuration data from a variety of sources, including the Amazon EC2 API, Amazon EC2 instance metadata, and Amazon CloudWatch, and shares it with SAP applications, to help monitor and improve the performance of business transactions.

• Any AWS account that you use for running SAP systems must have an AWS support plan for either Business Support or Enterprise Support.

Deploying SAP Systems on AWS

The section describes different options available for provisioning AWS infrastructure and installing SAP systems on AWS.

Manual Deployment

The majority of SAP solutions supported on AWS can be installed by manually provisioning the required AWS infrastructure resources and then following the relevant, standard SAP installation documentation to install the SAP system on AWS.

Automated Deployment

AWS Quick Starts are built by AWS solutions architects and partners to help you deploy popular solutions on AWS, based on AWS best practices for security and high availability. These reference deployments implement key technologies automatically on the AWS Cloud, and eliminate many of the manual steps required for deployment. You can build your test or production environment in a few steps, and start using it immediately.

Prebuilt Images

Some SAP solutions are available on AWS as a prebuilt system image that contains a preinstalled and preconfigured SAP system. A prebuilt SAP system image enables you to rapidly provision a new SAP system without spending the time and effort required by a traditional manual SAP installation.

Prebuilt SAP system images are available from the following sources:

• AWS Marketplace
• SAP Cloud Appliance Library

<table>
<thead>
<tr>
<th>SAP solution</th>
<th>Deployment option(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP Business Suite (ERP, CRM, etc.)</td>
<td>Manual</td>
</tr>
<tr>
<td>SAP NetWeaver</td>
<td>Manual</td>
</tr>
<tr>
<td>SAP S/4HANA</td>
<td>Manual</td>
</tr>
<tr>
<td>SAP BW/4HANA</td>
<td>Manual</td>
</tr>
<tr>
<td>SAP HANA</td>
<td>Manual</td>
</tr>
<tr>
<td>SAP HANA, express edition</td>
<td>AWS Marketplace</td>
</tr>
</tbody>
</table>
Getting Assistance from APN Partners

There are AWS Partner Network (APN) partners who are experienced in deploying and operating SAP solutions, and can help you with your SAP workloads on AWS. For additional information see the following section.

Partner Services for SAP on AWS

The AWS Partner Network (APN) is a community of companies that offer a wide range of services and products on AWS. APN SAP partners can provide SAP-specific services to help you fully maximize the benefits of running SAP solutions on AWS.

Types of Partner Services and Solutions for SAP on AWS

- **Cloud assessment services** – Advisory services to help you develop an efficient and effective plan for your cloud adoption journey. Typical services include financial/TCO (total cost of ownership), technical, security and compliance, and licensing.
- **Proof-of-concept services** – Services to help you test SAP on AWS; for example: SAP ERP/ECC migration to SAP HANA or SAP S/4HANA, SAP Business Warehouse (BW) migration to SAP HANA or SAP BW/4HANA, SAP OS/DB migrations, new SAP solution implementation.
- **Migration services** – Services to migrate existing SAP environments or systems to AWS; for example: all-on-AWS SAP migrations (PRD/QAS/DEV), hybrid SAP migrations (QAS/DEV), single SAP system (e.g., SAP BW) migrations.
- **Managed services** – Managed services for SAP environments on AWS, including: AWS account and resource administration, OS administration/patching, backup and recovery, SAP Basis and SAP NetWeaver.
- **Packaged solutions** – Bundled software and service offerings from SAP Partners that combine SAP software, licenses, implementation, and managed services on AWS, such as SAP S/4HANA, SAP BusinessObjects BI, and many others.
- **ISV software solutions** – Partner software solutions for the migration, integration, and operation of SAP solutions on AWS; for example: system migration, high availability, backup and recovery, data replication, automatic scaling, disaster recovery.

How to Find Partner Services and Solutions for SAP on AWS

The AWS SAP Partner Services and Solutions Directory provides a centralized place to search, discover, and connect with trusted APN partners who offer solutions and services to help your business achieve faster time to value and maximize the benefits of running SAP solutions on AWS. The AWS SAP Partner Services and Solutions Directory can be found on the AWS website at the following path:

https://aws.amazon.com/sap -> Partners -> Find a Partner
SAP on AWS Planning

If you are an experienced SAP Basis or SAP NetWeaver administrator, there are a number of AWS-specific considerations relating to compute configurations, storage, security, management, and monitoring that will help you get the most out of your SAP environment on AWS. This section provides guidelines for achieving optimal performance, availability, and reliability, and lower total cost of ownership (TCO) while running SAP solutions on AWS.

SAP Notes

Before migrating or implementing an SAP environment on AWS, you should read and follow the relevant SAP notes. Start from SAP Note 1656099 for general information and follow the links to other relevant SAP notes (SAP One Support Launchpad access required).

SAP on AWS Architectures

This section describes the two primary architectural patterns for SAP on AWS: all systems on AWS and hybrid.

All-on-AWS Architecture

With the SAP All-on-AWS architecture, all systems and components of your SAP environment are hosted on AWS. Example scenarios of such an architecture include:

- Implementation of a complete, new SAP environment on AWS
- Migration of a complete, existing SAP environment to AWS
Figure 3 depicts an SAP all-on-AWS architecture. The SAP environment running on AWS is integrated with on-premises systems and users via a VPN connection or a dedicated network connection via AWS Direct Connect. SAProuter is deployed in a public subnet and assigned a public IP address that is reachable from the internet to enable integration with the SAP OSS network via a virtual network computing (VNC) connection. A network address translation (NAT) gateway enables instances in the private subnet to connect to the internet or other AWS services, but prevents instances from receiving inbound traffic that is initiated by someone on the internet. For additional information, see the Configuring Network and Connectivity (p. 12) section.

Figure 3: SAP all-on-AWS architecture

Hybrid AWS Architecture

With an SAP hybrid AWS architecture, some SAP systems and components are hosted on your on-premises infrastructure and others are hosted on the AWS infrastructure. Example scenarios of such an architecture include:

- Running SAP test, trial, training, proof-of-concept (PoC), and similar systems on AWS
- Running non-production SAP landscapes (for example, DEV and QAS) on AWS, integrated with an SAP production landscape running on premises
- Implementing a new SAP application on AWS and integrating it with an existing SAP on-premises environment

Figure 4 depicts an SAP hybrid AWS architecture with SAP DEV and QAS landscapes and SAP test, training, and PoC systems running on AWS. These systems are integrated with SAP systems and users on the corporate network. Connectivity between the VPC and the corporate network is provided with either a VPN connection or an AWS Direct Connect connection. The existing SAProuter and SAP Solution Manager running on the corporate network are used to manage the SAP systems running within the VPC.
Choosing an AWS Region and Availability Zone

See the AWS Global Infrastructure (p. 5) section of this guide for information about AWS Regions and Availability Zones.

Choosing a Region

When choosing the AWS Region to deploy your SAP environment in, consider the following factors:

- Proximity to your on-premises data center(s), systems, and end users to minimize network latency.
- Data residency and regulatory compliance requirements.
- Availability of the AWS products and services you plan to use in the region. For a detailed list of AWS products and services by region, see the Region Table on the AWS website.
- Availability of the EC2 instance types you plan to use in the region. To view AWS Region availability for a specific instance type, see the Amazon EC2 Instance Types for SAP webpage.

Choosing an Availability Zone

No special considerations are required when choosing an Availability Zone for your SAP deployment on AWS. All SAP applications (SAP ERP, CRM, SRM, and so on) and systems (SAP database system, SAP Central Services system, and SAP application servers) should be deployed in the same Availability Zone. If high availability (HA) is a requirement, use multiple Availability Zones. For more information about high availability, see the SAP on AWS High Availability Guide.

Network and Connectivity

Amazon VPC

Amazon VPC enables you to define a virtual network in your own, logically isolated area within the AWS Cloud. You can launch your AWS resources, such as instances, into your VPC. Your VPC closely resembles
a traditional network that you might operate in your own data center, with the benefits of using the AWS scalable infrastructure. You can configure your VPC; you can select its IP address range, create subnets, and configure route tables, network gateways, and security settings. You can connect instances in your VPC to the internet. You can connect your VPC to your own corporate data center, and make the AWS Cloud an extension of your data center. To protect the resources in each subnet, you can use multiple layers of security, including security groups and network access control lists. For more information, see the Amazon VPC User Guide.

For detailed instructions for setting up and configuring a VPC, and connectivity between your network and VPC, see the Amazon VPC documentation.

**Network Connectivity Options**

Multiple options are available to provide network connectivity between your on-premises users and systems with your SAP systems running on AWS, including a direct internet connection, hardware VPN, and private network connection.

**Direct Internet Connection**

The quickest and simplest way to connect to your SAP systems running on AWS involves using a VPC with a single public subnet and an internet gateway to enable communication over the internet. For additional information, see Scenario 1: VPC with a Public Subnet Only in the Amazon VPC User Guide.

Use cases: Most suitable for SAP demo, training, and test type systems that do not contain sensitive data.

**Site-to-Site / Hardware VPN**

AWS Site-to-Site VPN extends your data center or branch office to the cloud via Internet Protocol security (IPsec) tunnels, and supports connecting to both virtual private gateways and AWS Transit Gateway. You can optionally run Border Gateway Protocol (BGP) over the IPsec tunnel for a highly available solution. For additional information, see Adding a Hardware Virtual Private Gateway to your VPC in the Amazon VPC User Guide.

Use cases: Recommended for any SAP environments on AWS that require integration with on-premises users and systems.

**Client VPN**

AWS Client VPN provides a fully-managed VPN solution that can be accessed from anywhere with an internet connection and an OpenVPN-compatible client. It is elastic, automatically scales to meet your demand, and enables your users to connect to both AWS and on-premises networks. AWS Client VPN seamlessly integrates with your existing AWS infrastructure, including Amazon VPC and AWS Directory Service, so you don’t have to change your network topology.

Use cases: Provides quick and easy connectivity to your remote workforce and business partners.

**Private Network Connection**

AWS Direct Connect makes it easy to establish a dedicated network connection from your premises to AWS. Using AWS Direct Connect, you can establish private connectivity between AWS and your data center, office, or co-location environment. In many cases, this can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than internet-based connections. For additional information, see the AWS Direct Connect User Guide.

Use cases: Recommended for customers who require greater bandwidth and lower latency than possible with a hardware VPN.

For additional information about the different Amazon VPC connectivity options, see the Amazon Virtual Private Cloud Connectivity Options whitepaper.
Following Security Best Practices

In order to provide end-to-end security and end-to-end privacy, AWS builds services in accordance with security best practices, provides appropriate security features in those services, and documents how to use those features. In addition, AWS customers must use those features and best practices to architect an appropriately secure application environment. Enabling customers to ensure the confidentiality, integrity, and availability of their data is of the utmost importance to AWS, as is maintaining trust and confidence.

Shared Responsibility Environment

There is a shared responsibility model between you as the customer and AWS. AWS operates, manages, and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the services operate. In turn, you assume responsibility and management of the guest operating system (including updates and security patches), other associated application software, Amazon VPC setup and configuration, as well as the configuration of the AWS-provided security group firewall. For additional information on AWS security, visit the AWS Cloud Security page and review the various Security Resources available there.

Amazon VPC

The foundation for security of an SAP environment on AWS is the use of Amazon VPC for providing the overall isolation. Amazon VPC includes security details that you must set up to enable proper access and restrictions for your resources. Amazon VPC provides features that you can use to help increase and monitor the security for your VPC:

- **Security groups** act as a firewall for associated EC2 instances, controlling both inbound and outbound traffic at the instance level.
- **Network access control lists (ACLs)** act as a firewall for associated subnets, controlling both inbound and outbound traffic at the subnet level.
- **Route tables** consist of a set of rules, called routes, that determine where network traffic is directed. Each subnet in your VPC must be associated with a route table; the table controls the routing for the subnet.
- **Flow logs** capture information about the IP traffic going to and from network interfaces in your VPC.

For detailed documentation about how to set up and manage security within a VPC, see the Security section of the Amazon VPC User Guide.

The Modular and Scalable VPC Quick Start provides a networking foundation based on AWS best practices for your AWS Cloud infrastructure. It builds a VPC environment with public and private subnets where you can launch AWS services and other resources. Use this Quick Start as a building block for your own deployments. You can scale it up or down as needed, and add other infrastructure components and software layers to complete your AWS environment.

EC2 Instance Types for SAP

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instance types comprise varying combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your applications. Each instance type includes one or more instance sizes, allowing you to scale your resources to the requirements of your target workload.

SAP systems deployed on AWS that will require support from SAP must be run on an EC2 instance type that has been certified with SAP. This section describes where you can find details about the EC2 instance types that have been certified with SAP and additional information for specific SAP solutions.
SAP NetWeaver-based Solutions

SAP solutions based on the SAP NetWeaver platform and that use SAP Application Performance Standard (SAPS) for sizing must be run on a specific subset of EC2 instance types in order to receive support from SAP Support. For details, see:

- SAP Note 1656099
- Amazon EC2 Types for SAP

SAP HANA

The SAP HANA platform and SAP solutions that run on top of an SAP HANA database—for example, SAP Suite on HANA, SAP S/4HANA, SAP Business Warehouse (BW) on HANA, SAP BW/4HANA—require specific EC2 instance types that have been certified for SAP HANA. For details, see:

- SAP HANA-certified IaaS platforms
- Amazon EC2 Types for SAP
- Smaller X1e instances for SAP HANA non-production workloads

SAP Business One, version for SAP HANA

For information about the EC2 instance types that are certified for SAP Business One, version for SAP HANA, see:

- SAP Note 2058870
- SAP Business One on AWS

Operating Systems

Supported Operating Systems

EC2 instances run on 64-bit virtual processors based on the Intel x86 instruction set. The following 64-bit operating systems and versions are available and supported for SAP solutions on AWS.

- SUSE Linux Enterprise Server (SLES)
- SUSE Linux Enterprise Server for SAP Applications (SLES for SAP)
- Red Hat Enterprise Linux (RHEL)
- Red Hat Enterprise Linux for SAP Solutions (RHEL for SAP)
- Microsoft Windows Server
- Oracle Enterprise Linux

For additional information regarding SAP-supported operating systems on AWS, see SAP Note 1656250.

SLES for SAP and RHEL for SAP

SUSE and Red Hat offer SAP-specific versions of their operating systems that provide the following benefits:

- Configuration and tuning for SAP
- Extended release support
• High availability extension for SAP
• Dedicated support channel

**Note**
Because of these benefits, we strongly recommend using SLES for SAP or RHEL for SAP with High Availability (HA) and Update Services (US) for your SAP on AWS deployments.

To learn more about SUSE’s and Red Hat’s operating system versions for SAP, see the following information on the SLES and Red Hat websites.

**SLES for SAP**
• General information
• SUSE on AWS for SAP Applications
• SUSE Public Cloud Program – Bring Your Own Subscription (FAQ)

**RHEL for SAP**
• General information
• Red Hat in the Public Cloud
• Red Hat Cloud Access
• How to Locate Red Hat Cloud Access Gold Images on AWS EC2
• What is the Difference between Red Hat Cloud Access and Red Hat Enterprise Linux On-Demand Subscriptions in the public cloud?

**Operating System Licenses**

These operating system licensing options are available for SAP systems on AWS:

• **On-demand** – The operating system software and license are bundled in an Amazon Machine Image (AMI). The fee for the operating system license is included in the On-Demand Instance hourly fee or Reserved Instance fee for the instance type.
• **Bring Your Own License/Subscription (BYOL)** – Bring your existing operating system license or subscription to the AWS Cloud.
• **AWS Marketplace** – Purchase operating system licenses and subscriptions from AWS Marketplace.

The following table lists the licensing options available for each operating system and version. To learn more about each option, follow the link in the table.

<table>
<thead>
<tr>
<th>Operating system</th>
<th>License/subscription options</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLES</td>
<td>On-demand</td>
</tr>
<tr>
<td>SLES for SAP</td>
<td>AWS Marketplace</td>
</tr>
<tr>
<td>RHEL</td>
<td>On-demand</td>
</tr>
<tr>
<td>RHEL for SAP with HA and US</td>
<td>AWS Marketplace</td>
</tr>
<tr>
<td>Windows</td>
<td>On-demand</td>
</tr>
<tr>
<td>Oracle Linux</td>
<td>BYOL</td>
</tr>
</tbody>
</table>
Databases

Supported Databases

All the database platforms and versions supported by SAP for an on-premises infrastructure are also supported by SAP on AWS. For details about the databases supported with specific SAP solutions on AWS, see SAP Note 1656099.

Database Installation and Administration

Customer-Managed Database on Amazon EC2

The majority of SAP solutions use a customer-managed model on Amazon EC2. Installation, configuration, administration, and backup and recovery of the database are done by either the customer or a partner.

The following SAP solutions use a self-managed database model on Amazon EC2:

- SAP Business Suite and SAP NetWeaver-based applications
- SAP HANA
- SAP S/4HANA
- SAP BW/4HANA
- SAP BusinessObjects BI
- SAP Business One

Amazon RDS

Amazon Relational Database Service (Amazon RDS) is a managed service that makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity, while managing time-consuming database administration tasks, freeing you up to focus on your applications and business. Amazon RDS is currently supported for the following SAP solutions:

- SAP BusinessObjects BI
- SAP Commerce (previously known as SAP Hybris Commerce)

Amazon RDS

Amazon Aurora (Aurora) is a MySQL and PostgreSQL-compatible relational database built for the cloud. It combines the performance and availability of traditional enterprise databases with the simplicity and cost-effectiveness of open source databases. Aurora MySQL is currently supported for the following SAP solution:

- SAP Commerce (previously known as SAP Hybris Commerce)

Database Licenses

These database licensing options are available for SAP systems on AWS:

- **On-demand** – The database software and license are bundled in an Amazon Machine Image (AMI). The fee for the database license is included in the On-Demand Instance hourly fee or Reserved Instance fee for the instance type.
- **Bring Your Own License (BYOL)** – Bring your existing database licenses to the AWS Cloud.
• **AWS Marketplace** – Purchase database software and licenses from AWS Marketplace.

The following table lists the licensing options available on AWS for each database. For additional information, follow the links in the Licensing options column.

<table>
<thead>
<tr>
<th>Database</th>
<th>Licensing options</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP HANA</td>
<td>BYOL</td>
</tr>
<tr>
<td>SAP HANA, express edition</td>
<td>AWS Marketplace</td>
</tr>
<tr>
<td>SAP Adaptive Server Enterprise (ASE) (SAP ASE)</td>
<td>BYOL</td>
</tr>
<tr>
<td>Microsoft SQL Server</td>
<td>BYOL*</td>
</tr>
<tr>
<td>IBM DB2</td>
<td>BYOL</td>
</tr>
<tr>
<td>Oracle</td>
<td>BYOL</td>
</tr>
<tr>
<td>Amazon Aurora</td>
<td>On-demand</td>
</tr>
</tbody>
</table>

* SQL Server runtime licenses purchased from SAP require either Microsoft Software Assurance or Amazon EC2 Dedicated Hosts to bring these licenses to AWS. For additional information, see:

- SAP Note 2139358 - Effect of changes in licensing terms of SQL Server
- Microsoft Licensing on AWS

**SAP Installation Media**

The majority of SAP solutions on AWS use a bring-your-own-software model. There are two primary options for copying SAP installation media to AWS:

- **Download from the SAP Software Download Center to Amazon EC2.** From your EC2 instance, connect to the SAP Software Download Center and download the required installation media. This option will most likely be the fastest method for getting SAP installation media to AWS, because EC2 instances have very fast connections to the internet. You can create a dedicated Amazon EBS volume to store installation media, and then attach the volume to different instances as needed. You can also create a snapshot of the Amazon EBS volume and create multiple volumes that you can attach to multiple instances in parallel.
- **Copy from your network to Amazon EC2.** If you already have the required SAP installation media downloaded to a location on your network, you can copy the media from your network directly to an EC2 instance.

**SAProuter and SAP Solution Manager**

The following sections describe options for SAProuter and SAP Solution Manager when running SAP solutions on AWS.

**For SAP All-on-AWS Architecture**

When setting up an SAP environment on AWS, you will need to set up an SAP Solution Manager system and SAProuter with a connection to the SAP support network, as you would with any infrastructure. See the all-on-AWS architecture diagram (Figure 3 (p. 11)) for an illustration.
When setting up the SAProuter and SAP support network connection, follow these guidelines:

- Launch the instance that the SAProuter software is installed on into a public subnet of the VPC and assign it an Elastic IP address.
- Create a specific security group for the SAProuter instance with the necessary rules to allow the required inbound and outbound access to the SAP support network.
- Use the Secure Network Communications (SNC) type of internet connection. For more information, see https://service.sap.com/internetconnection.

**For SAP Hybrid AWS Architecture**

When using AWS as an extension of your IT infrastructure, you can use your existing SAP Solution Manager system and SAProuter that are running in your data center to manage SAP systems running on AWS within a VPC. See the hybrid architecture diagram (Figure 4 (p. 11)) for additional information.

**Additional Reading**

**SAP on AWS technical documentation**

- SAP HANA Environment Setup on AWS
- SAP HANA on AWS Operations Guide
- SAP NetWeaver Environment Setup for Linux on AWS

**Document Revisions**

<table>
<thead>
<tr>
<th>Date</th>
<th>Change</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>May 2019</td>
<td>Update</td>
<td>Changes throughout guide</td>
</tr>
<tr>
<td>August 2018</td>
<td>Initial publication</td>
<td>–</td>
</tr>
</tbody>
</table>
AWS Data Provider for SAP

Installation and Operations Guide

AWS Data Provider Version 3.0

SAP specialists, Amazon Web Services (AWS)

Last updated (p. 60): December 2020

The Amazon Web Services (AWS) Data Provider for SAP is a tool that collects performance-related data from AWS services. It makes this data available to SAP applications to help monitor and improve the performance of business transactions. The AWS Data Provider for SAP uses operating system, network, and storage data that is most relevant to the operation of the SAP infrastructure. Its data sources include Amazon Elastic Compute Cloud (Amazon EC2) and Amazon CloudWatch. This guide provides installation, configuration, and troubleshooting information for the AWS Data Provider for SAP on both Linux and Windows.

About this Guide

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the AWS Cloud. For the other guides in the series, ranging from overviews to advanced topics, see the SAP on AWS Technical Documentation home page.

Introduction

Many organizations of all sizes are choosing to host key SAP systems in the Amazon Web Services (AWS) Cloud. With AWS, you can quickly provision an SAP environment. Additionally, the elastic nature of the AWS Cloud enables you to scale computing resources up and down as needed. As a result, your business can dedicate more resources (both people and funds) to innovation.

Many SAP systems operate daily business transactions and are critical to business functions. As an SAP customer, you need the ability to track and troubleshoot the performance of these transactions. The AWS Data Provider for SAP is a tool that collects key performance data on an Amazon Elastic Compute Cloud (Amazon EC2) instance that SAP applications can use to monitor transactions built by SAP. The data is collected from a variety of sources within your AWS Cloud operating environment, including Amazon EC2 and Amazon CloudWatch. This data includes information about the operating system, network, and storage that is relevant to your SAP infrastructure. Data from the AWS Data Provider for SAP is read by the SAP Operating System Collector (SAPOSCOL) and the SAP CIM Provider.

Figure 1 provides a high-level illustration of the AWS Data Provider for SAP, its data sources, and its outputs.
The purpose of this guide is to help you:

- Understand the technical requirements and components necessary to install and operate the AWS Data Provider for SAP.
- Install the AWS Data Provider for SAP.
- Understand the update process for the AWS Data Provider for SAP.
- Troubleshoot installation issues.

**Upgrading from Earlier Versions**

The current version of the AWS Data Provider for SAP is version 3.0. Version 3.0 is not compatible with version 2.9 or older. Version 3.0 also requires a new AWS Identity and Access Management (IAM) policy. For a list of changes in each version, see Appendix C (p. 54). If you have an earlier version installed, you must uninstall it before installing the current version.

**Note on versions 2.9**

If you've already installed and customized the AWS Data Provider for SAP version 2.9 or older, save your custom `proxy.properties` file in the temp directory of your operating system. Then uninstall version 2.9 or older and install version 3.0 which will look for the `proxy.properties` file in the temp directory by default and use it for the new installation.

Uninstalling the data provider does not require SAP downtime and can be done online. The only impact will be a gap in metric monitoring information for the time there was in provider installed on the system.

See Appendix D (p. 56) on how to uninstall older Data Provider agents.

**Technical Requirements**

Before creating an SAP instance, ensure that the following technical requirements are met.
Amazon VPC Network Topologies

You need to deploy SAP systems that receive information from the AWS Data Provider for SAP within an Amazon Virtual Private Cloud (Amazon VPC). You can use one of the following network topologies to enable routing to internet-based endpoints:

- The first topology configures routes and traffic directly to the AWS Cloud through a NAT gateway within an Amazon VPC (see Figure 2). For more information about internet gateways, see the AWS documentation.

![Figure 2: Connection to the AWS Cloud via an internet gateway](image)

- A second topology routes traffic from the Amazon VPC, through your organization’s on-premises data center, and back to the AWS Cloud (see Figure 3). For more information about this topology, see the AWS documentation.

![Figure 3: Connection to the AWS Cloud via an on-premises data center](image)

Amazon VPC Endpoints

Create endpoints for the following services that the data provider uses:

- Monitoring
- Amazon EC2
To create data endpoints in the AWS console, use the following procedure for each of the two endpoints:

1. Sign in to the Amazon VPC console, navigate to **Endpoints**, and select **Create Endpoint**.

![Figure 4: Creating a VPC endpoint](image-url)

2. On the next screen, search for the service name, then select the appropriate VPC and route table, and select **Create Endpoint**.
3. After creating all three endpoints you should see them in your list of endpoints as shown below:

<table>
<thead>
<tr>
<th>Name</th>
<th>Endpoint ID</th>
<th>VPC ID</th>
<th>Service name</th>
<th>Endpoint type</th>
</tr>
</thead>
<tbody>
<tr>
<td>vpc-01a01b03c02b3c02</td>
<td>vpc-001a0b02c02b3c02</td>
<td>private VPC</td>
<td>com.amazonaws.us-east-1.monitoring</td>
<td>Interface</td>
</tr>
<tr>
<td>vpc-01a01b03c02b3c02</td>
<td>vpc-001a0b02c02b3c02</td>
<td>private VPC</td>
<td>com.amazonaws.us-east-1.ec2</td>
<td>Interface</td>
</tr>
</tbody>
</table>

**Figure 6: VPC endpoints created**

**IAM Roles**

You need to grant the AWS Data Provider for SAP read-only access to the Amazon CloudWatch, Amazon Simple Storage Service (Amazon S3), and Amazon EC2 services so that you can use their APIs. You can do this by creating an AWS Identity and Access Management (IAM) role for your Amazon EC2 instance and attaching a permissions policy.

Use the following procedure to create an IAM role and grant permissions to your Amazon EC2 instance:

1. Sign in to the AWS Management Console and open the IAM console.
2. In the navigation pane, select Roles, and select Create role.
3. Choose the AWS service role type, and select EC2.
4. Select EC2 as the use case, and select Next Permissions.
5. Select **Create Policy**, and select **JSON**.

![Create policy](image)

**Figure 9: Create your own policy**

6. Copy and paste the following policy into the input field, replace all existing text, and select **Review Policy**.

**Note**
Customers who are running their Amazon EC2 instances in a Chinese AWS region need to change the **Resource line** listed on the next page to the following:

```json
arn:aws-cn:s3:::aws-sap-data-provider-china/config.properties
```

```json
{
    "Version": "2012-10-17",
    "Statement": [
    
    ]
}
```
"Sid": "VisualEditor0",
"Effect": "Allow",
"Action": [
  "EC2:DescribeInstances",
  "cloudwatch:GetMetricStatistics",
  "EC2:DescribeVolumes"
],
"Resource": "*"
},
{
"Sid": "VisualEditor1",
"Effect": "Allow",
"Action": "s3:GetObject",
"Resource": [
  "arn:aws:s3:::aws-sap-data-provider/config.properties"
]
}
}

7. Provide a **Name** and **Description** for the role, and select **Create Policy**.

![Image of IAM console](image)

**Figure 10: Provide name and description**

8. Select **Create Policy**. The IAM console confirms the new policy with a message similar to the following.

![Confirmation message](image)

**Figure 11: Policy confirmation**
9. Navigate to the **Create Role** page, refresh the screen, search for the newly created role, and select the policy.

![Create role](image)

**Figure 12: Policy confirmation**

10. Select **Next:Tags**.

11. Add any tags if needed, otherwise select **Next:Review**.

12. Provide a name for the Role and select **Create Role**.

![Create role](image)

**Figure 13: Role confirmation**

---

**Installing the AWS Data Provider for SAP**

The AWS Data Provider for SAP runs as a service that automatically starts at boot and collects, aggregates, and exposes metrics to the SAP host agent. Metrics are sourced from a variety of providers that pull metrics from the relevant areas of the platform. The AWS Data Provider for SAP is designed to continue operating, regardless of whether its providers have connectivity or permissions to access the AWS service metrics they are requesting. Providers that cannot reach the metrics they are harvesting return blank values.

For example, if your Amazon EC2 instance does not have an IAM role associated with it that grants explicit access to the Amazon CloudWatch **GetMetricStatistics** API, the CloudWatch provider will be
unable to perform the **GetMetricStatistics** action on the Amazon EC2 instance and will return blank values.

The provider needs to be installed on each SAP production system in order to be eligible for SAP support. You can only install one instance of the provider at a time on a system.

The AWS Data Provider for SAP is designed to automatically update itself so that it can provide you with the most current metrics. When the AWS Data Provider for SAP starts up, a built-in update service retrieves the latest versions of its components and metric definitions from an AWS managed Amazon S3 bucket. If the AWS Data Provider for SAP cannot access the update service, it will continue to run as-is.

## Installing on Linux

On Linux the data provider is delivered as an RPM package.

### SUSE Linux Enterprise Server

To install the AWS Data Provider for SAP on SUSE Linux Enterprise Server (SLES) download the following files:

- **Standard**: `aws-sap-dataprovider-sles.x86_64.rpm` and GPG Key
- **China**: `aws-sap-dataprovider-sles.x86_64.rpm` and GPG Key

The files are identical but AWS offers these two location options due to possible connectivity issues when working from China.

For customers who are behind a firewall and can't use a proxy there is a standalone version of the Data Provider. This will not connect to S3 for updates and will run as a self-contained package. It also means that it needs to be updated manually.

- **Standalone Standard**: `aws-sap-dataprovider-sles-standalone.x86_64.rpm` and GPG Key
- **Standalone China**: `aws-sap-dataprovider-sles-standalone.x86_64.rpm` and GPG Key

The GPG key is only required on SUSE and it is used by SUSE's package manager to validate the origin of the RPM package.

To install the data provider run the following commands:

```bash
wget https://<url to rpm package>
wget https://<url to GPG key>
rpm --import RPM-GPG-KEY-AWS
zypper install -y <rpm package>
```

**Example:**

```bash
wget https://aws-sap-data-provider.s3.amazonaws.com/Installers/aws-sap-dataprovider-sles.x86_64.rpm
rpm --import RPM-GPG-KEY-AWS
zypper install -y aws-sap-dataprovider-sles.x86_64.rpm
```

If you don't have transparent HTTP/HTTPS access to the internet, before starting the installer place a text file called `proxy.properties` in `/tmp`. 

---

28
The file should have the following contents:

```
# proxy.properties
# used to set web proxy settings for the AWS Data Provider for SAP
# Https is the only supported proxy method
# Blank values for everything means no proxy set
https.proxyHost=
https.proxyPort=
https.proxyDomain=
https.proxyUsername=
https.proxyPassword=
```

The file should have the following contents with values specified for the parameters that you want to add (empty means no setting):

When the RPM package is installed you will see output onscreen, such as the installer pulling down the required files it needs, as shown below.

```
Figure 14: AWS Data Provider for SAP update service
```

When it's completed, you will see the agent start as a daemon.

```
Figure 15: Completing the installation on Linux
```

Verify that the service is running by calling `netstat -ant` to determine if the listener is running on localhost port 8888.

```
Figure 16: Verifying the installation on Linux
```

You should also view the log files at `/var/log/aws-dataprovider/messages` to ensure the daemon has the appropriate connectivity and authorization to access the required metrics.
At startup, the monitoring agent runs three sets of diagnostics:

- The AWS connectivity diagnostic ensures network connectivity to Amazon S3 for obtaining automatic updates to the AWS Data Provider for SAP.
- The second diagnostic tests for authorization to access CloudWatch. This authorization requires assigning an IAM role to the Amazon EC2 instance you are running on with an IAM policy that allows access to CloudWatch. For details, see IAM Roles (p. 24), earlier in this guide.
- The third diagnostic tests for authorization to access Amazon EC2, which also requires an IAM role associated with the Amazon EC2 instance.

The AWS Data Provider for SAP is designed to run with or without connectivity, but you can't obtain updates without connectivity. Amazon CloudWatch and Amazon EC2 will return blank values if you don't have the proper authorizations in place.

You can also call the AWS Data Provider for SAP directly to view the metrics. Calling `wget http://localhost:8888/vhostmd` returns a file of metrics. You can look inside the file to see the metrics that were returned, as shown here.
The AWS Data Provider for SAP now starts automatically each time the operating system starts. You can also manually stop and restart the AWS Data Provider for SAP with the following command, which depends on your operating system version:

- **SLES 11, Oracle Linux 6, and Red Hat Linux 6:**
  
  ```bash
  service aws-dataprovider [start|stop]
  ```

- **SLES 12, Oracle Linux 7, and Red Hat Linux 7:**
  
  ```bash
  systemctl [start|stop] aws-dataprovider
  ```

### Installing on Red Hat and Oracle Enterprise Linux

For Red Hat and OEL (Oracle Enterprise Linux) the installation steps are the same as described for SLES above but the RPM file and command to install the RPM package differs.

- **Default:** `aws-sap-dataprovider-rhel.x86_64.rpm`
- **China:** `aws-sap-dataprovider-rhel.x86_64.rpm`

The two files are identical but AWS offers these two location options due to possible connectivity issues when working from China.
For customers who are behind a firewall and can’t use a proxy there is a standalone version of the Data Provider. This will not connect to S3 for updates and will run as a self-contained package. It also means that it needs to be updated manually.

- **Standalone Standard**: aws-sap-dataprovier-rhel-standalone.x86_64.rpm
- **Standalone China**: aws-sap-dataprovier-rhel-standalone.x86_64.rpm

To install the data provider run the following commands:

```bash
wget https://<url to rpm package>
yum -y install <rpm package>
```

**Example:**

```bash
wget https://aws-sap-data-provider.s3.amazonaws.com/Installers/aws-sap-dataprovier-rhel.x86_64.rpm
yum -y install aws-sap-dataprovier-rhel.x86_64.rpm
```

### Installing on Windows

On Windows the installer is delivered in the form of an NSIS (Nullsoft Scriptable Install System) executable.

1. Open a web browser and download the installer:
   - **Default**: aws-data-provider-installer-win-x64.exe
   - **China**: aws-data-provider-installer-win-x64.exe

   The two files are identical but AWS offers these two location options due to possible connectivity issues when working from China.

   For customers who are behind a firewall and can’t use a proxy there is a standalone version of the Data Provider. This will not connect to S3 for updates and will run as a self-contained package. It also means that it needs to be updated manually.

2. If you don’t have transparent HTTP/HTTPS access to the internet, before starting the installer place a text file called `proxy.properties` in the Windows temp directory (%TEMP%).

   The file should have the following contents with values specified for the parameters that you want to add (empty means no setting):

   ```
   # proxy.properties
   # used to set web proxy settings for the AWS Data Provider for SAP
   # Https is the only supported proxy method
   # Blank values for everything means no proxy set
   https.proxyHost=  
   https.proxyPort=  
   https.proxyDomain=  
   https.proxyUsername=  
   https.proxyPassword=  
   ```

   The installer will copy the file to the installation directory and then delete it.

3. Open **Proxy Settings** from the Windows start menu.
Figure 19: Proxy Settings in Windows

4. In the Manual proxy setup section of the Proxy window:
   a. Click Use a proxy server to enable it.
   b. Enter the host name or IP address of the proxy server.
   c. Enter the port number on which the proxy server listens.
   d. Click Save.
Manual proxy setup

Use a proxy server for Ethernet or Wi-Fi connections. These settings don’t apply to VPN connections.

Use a proxy server

- **On**

Address

Port

Use the proxy server except for addresses that start with the following entries. Use semicolons (;) to separate entries.

☑ Don’t use the proxy server for local (intranet) addresses

Save

Figure 20: Manual Proxy Setup
5. Run the installer by double clicking the exe and wait for it to finish.
6. Check the installation:

- When the script has completed, the software is installed in the C:\Program Files\Amazon\DataProvider directory.
- The installation also creates and starts a Windows service named **AWS Data Provider for SAP**.
- To ensure that the service is running, open a web browser and enter http://localhost:8888/vhostmd. If it’s running, it returns a page with metrics from the AWS Data Provider for SAP.

7. Configure the AWS Data Provider for SAP to use proxies:

- Stop the Windows service **AWS Data Provider for SAP**.
- Enter the required proxy information into this file: C:\Program Files\Amazon\DataProvider\proxy.properties.
- Restart the Windows service, **AWS Data Provider for SAP**.
- When the service starts, it performs an update of the agent, and then works in standard mode.

8. Verify that the service is running by calling `netstat -ant` from a command window or from a Windows PowerShell script to determine if the listener is running on localhost port 8888.

9. Navigate to the Windows event log, and find the application log for startup events from the AWS Data Provider for SAP. You should look at the diagnostics.
Figure 23: Checking diagnostics on Windows

At startup, the monitoring agent runs three sets of diagnostics:

- The AWS connectivity diagnostic ensures network connectivity to Amazon S3 for obtaining automatic updates to the AWS Data Provider for SAP.
- The second diagnostic tests for authorization to access CloudWatch, which requires assigning an IAM role to the EC2 instance you are running on with an IAM policy that allows access to CloudWatch. For details, see IAM Roles (p. 24), earlier in this guide.
- The third diagnostic tests for authorization to access Amazon EC2, which also requires an IAM role associated with the Amazon EC2 instance.

The AWS Data Provider for SAP is designed to run with or without connectivity, but you can't obtain updates without connectivity. If you don't have the proper authorizations in place, Amazon CloudWatch and Amazon EC2 return blank values.

You can also call the AWS Data Provider for SAP directly from your web browser to view metrics, as shown in Figure 24.
The AWS Data Provider for SAP now starts automatically each time the operating system starts. You can also manually stop and restart the AWS Data Provider for SAP, just as you would stop and restart any other Windows service.

In order to configure proxy settings, you can place a customized `proxy.properties` file in Windows's temp directory, which is designated by the windows system variable `%TEMP%`. 
Updating the AWS Data Provider for SAP

AWS recommends to always uninstall an existing version of the AWS DataProvider for SAP before installing a new version. If you are running DataProvider version 2.9 or lower, you can’t update to version 3.0, and will have to uninstall the old version first.

If you’re using a proxy configuration, you should preserve the `proxy.properties` file. The easiest way to do this is by copying it into the temp directory of your operating system where version 3.0 data provider installers will look for the file and use it during installation.

Examples:

Linux

```bash
cp /usr/local/ec2/aws-dataprovider/proxy.properties /tmp
```

Windows:

```bash
copy “C:\Program Files\Amazon\DataProvider\proxy.properties” “%TEMP%”
```

Uninstalling on Linux

1. Log in to Linux as a superuser, like root.
2. Stop and remove the Data Provider using the following command.

   SLES

   ```bash
   zypper remove -y aws-sap-dataprovider
   ```

   RHEL / OEL

   ```bash
   yum -y erase aws-sap-dataprovider
   ```

Uninstalling on Windows

1. Run the uninstaller.

   ```bash
   C:\Program Files\Amazon\DataProvider\uninstall.exe
   ```

2. When prompted, choose Uninstall.
Troubleshooting

This section provides help to analyze installation problems.

Troubleshooting on Linux

Problem: The installation failed, and I’m not sure if my files are in a consistent state.

Stop and remove the Data Provider with the following command.

SLES:

```
- remove -y aws-sap-dataprovider
```

RHEL / OEL:

```
yum -y erase aws-sap-dataprovider
```

Problem: The AWS Data Provider for SAP failed to start at the end of the installation process.

Check the log files in `/var/log/aws-dataprovieder` for hints on what is not going as expected. If needed uninstall and reinstall the Data Provider. If reinstalling the AWS Data Provider for SAP doesn't solve the problem, you can gather debug information about the AWS Data Provider for SAP by editing the `/usr/local/ec2/aws-dataprovieder/bin/aws-dataprovieder` file.
Now if you run service `aws-dataprocessor-start` or `systemctl start aws-dataprocessor`, you will get a lot of debugging output that might help you diagnose the root cause of the problem.

Figure 28: Debugging information on Linux
Problem: When I looked at my logs I noticed that my installation failed all diagnostics.

```
14:32:15.862 I 08001 ** Running Diagnostics **
14:32:15.862 I 08002 Diagnostic : AWS Connectivity
14:33:19.362 W 08003 Diagnostic : Failed
14:33:19.362 I 08006 Diagnostic : Amazon CloudWatch Connectivity & Access
14:33:19.515 W 08007 Diagnostic : Failed
14:39:51.516 I 0800A Diagnostic : EC2 API Connectivity & Access
14:39:54.542 W 0800B Diagnostic : Failed
14:39:54.542 I 0800E ** Diagnostics Complete **
```

Figure 29: Symptoms of internet connectivity problems on Linux

Failing all diagnostics indicates that there's a problem with your outbound connection to the internet. You can confirm this by pinging a well-known internet location, like www.amazon.com. The most common cause of routing issues is in the VPC network configuration, which needs to have either an internet gateway in place or a VPN connection to your data center with a route to the internet. For details, see Amazon VPC Network Topologies (p. 22), earlier in this guide.

Problem: When I looked at my logs I noticed that I don't have access to CloudWatch and Amazon EC2, but I did pass the first diagnostic for AWS connectivity.

```
14:38:57.467 I 08001 ** Running Diagnostics **
14:38:57.468 I 08002 Diagnostic : AWS Connectivity
14:38:58.182 I 08005 Diagnostic : Passed
14:38:58.182 I 08006 Diagnostic : Amazon CloudWatch Connectivity & Access
14:38:58.925 W 08007 Diagnostic : Failed
14:38:58.925 I 0800A Diagnostic : EC2 API Connectivity & Access
14:38:58.957 W 0800B Diagnostic : Failed
14:38:58.957 I 0800E ** Diagnostics Complete **
```

Figure 30: Symptoms of authorization issues on Linux

This is a clear indicator that you have an authorization issue when trying to access CloudWatch and Amazon EC2. The common cause for this problem is not having an IAM role associated with your instance that contains the IAM policy, as specified in IAM Roles (p. 24), earlier in this guide. You can quickly diagnose this issue by looking at the Amazon EC2 instance in question in the Amazon EC2 console and verifying the IAM role.
If the IAM role doesn't exist, then create it as specified in IAM Roles, which is described earlier in this guide.

If you do have an IAM role assigned to the instance, go to the IAM console, select the IAM role name, and then expand the policy. Verify that you have the required policy that is specified in IAM Roles (p. 24), earlier in this guide.

Troubleshooting on Windows

Problem: The installation failed, and I’m not sure if my files are in a consistent state.

Follow the procedure for the section called “Uninstalling on Windows” (p. 38), then the section called “Installing on Windows” (p. 32).
Problem: The AWS Data Provider for SAP failed to start at the end of the installation process.

Follow the procedure for the section called “Uninstalling on Windows” (p. 38), then the section called “Installing on Windows” (p. 32).

If reinstalling the AWS Data Provider for SAP doesn't solve the problem, you can gather debugging information about the AWS Data Provider for SAP by reviewing the log files in the C:\Program Files \Amazon\DataProvider directory.

These log files include an installation log, a log of the service installation, and the output of the AWS Data Provider for SAP itself.

![Log files on Windows]

Figure 33: Log files on Windows

Problem: I want to get more detailed log information from the data provider.

Start by stopping the data provider service.

![Stop Service on Windows]

Figure 34: Stop Service on Windows

Open the registry editor by clicking on the Windows logo in the bottom left and typing regedit and then clicking the option that is shown on screen:

![Start regedit]

Figure 35: Start regedit
In the registry, navigate to the key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Apache Software Foundation\Procrun 2.0\awsDataProvider\Start
```

![Figure 36: Logging setting](image)

The data provider accepts two log levels: INFO and FINE. FINE will generate more detailed logging which can be useful when debugging a problem. The recommendation is to set it back to INFO when you have finished troubleshooting to avoid the logs consuming unnecessary disk space.

**Problem: I want to reinstall the AWS Data Provider for SAP from scratch.**

Follow the procedure for the section called “Uninstalling on Windows” (p. 38), then the section called “Installing on Windows” (p. 32).

**Problem: When I looked at my logs, I noticed that my installation failed all diagnostics.**

```
14:32:15.862 I 08001  ** Running Diagnostics **
14:32:15.862 I 08002 Diagnostic : AWS Connectivity
14:33:19.362 W 08003 Diagnostic : Failed
14:33:19.362 I 08006 Diagnostic : Amazon CloudWatch Connectivity & Access
14:33:19.515 W 08007 Diagnostic : Failed
14:33:19.516 I 0800A Diagnostic : EC2 API Connectivity & Access
14:33:19.542 W 0800B Diagnostic : Failed
14:33:19.542 I 0800E ** Diagnostics Complete **
```

![Figure 37: Symptoms of internet connectivity problems on Windows](image)

Failing all diagnostics indicates that there’s a problem with your outbound connection to the internet. You can confirm this by pinging a well-known internet location, like www.amazon.com. The most common cause of routing issues is in the VPC network configuration, which needs to have either an internet gateway in place or a VPN connection to your data center with a route to the internet.
Problem: When I looked at my logs, I noticed that I don’t have access to CloudWatch and Amazon EC2, but I did pass the first diagnostic for AWS connectivity.

This is a clear indicator that you have an authorization issue when trying to access Amazon CloudWatch and Amazon EC2. The common cause for this problem is not having an IAM role associated with your instance that contains the IAM policy, as specified in IAM Roles (p. 24) earlier in this guide. You can quickly diagnose this issue by looking at the specific EC2 instance in the Amazon EC2 console and verifying the IAM role.

If the IAM role doesn’t exist, then create is as specified in IAM Roles, described earlier in this guide.

If you do have an IAM role assigned to the instance, go to the IAM console, select the IAM role name, and then choose Show. Verify that you have the required policy that is specified in IAM Roles (p. 24).
Appendix A: Customizing the AWS Data Provider for SAP

Some settings are hard coded in the AWS Data Provider for SAP. You can override existing settings or add new settings. For example, when AWS adds new instance types, you can add these to the AWS Data Provider for SAP configuration.

The AWS Data Provider for SAP creates a database by reading the configuration information from the following files, in this sequence:

- It reads the `config.properties` file from the JAR (Java Archive) file of the data provider application.
- It reads the file from `https://s3.amazonaws.com/aws-data-provider/config.properties`, which provides settings for Amazon EC2 instance types and Amazon Elastic Block Store (Amazon EBS) volume types. For example, when AWS releases new instance types, AWS updates this file. This file doesn't have to exist.
- It reads the file from the directory in which the proxy configuration file is located. This file is required only if a user wants to override or extend the current configuration. Default locations are:
  - On Linux: `/usr/local/ec2/aws-agent/config.properties`
  - On Windows: `C:\Program Files\Amazon\DataProvider\config`

**Syntax Rules for Configuration Files**

- The configuration files require a comma after the last value in every row.
- Spaces are not ignored in strings. The entire string between the commas, including any spaces, is accepted as the value.
- If there are multiple rows with the same instance type, the existing value for that type is overwritten.
- Capitalization in strings is case sensitive.
User-Configurable EC2 Instance Types

The AWS Data Provider for SAP maintains a database of all relevant Amazon EC2 instance types for SAP. Entries for EC2 instance types must be in a comma-separated list, as follows:

```
ec2type,i-type,cpu,core,threads,t-ecu,ecu,hthread,l-map,w-map,speed,p-ecu,
```

For example:

```
ec2type,r3.8xlarge,2,16,2,32,1,thread,eth0,lan2,10000,true,
```

where the following applies:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Content</th>
<th>Example</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>keyword</td>
<td>ec2type</td>
<td>—</td>
<td>String</td>
<td>A token to identify a record with an EC2 instance description</td>
</tr>
<tr>
<td>i-type (instance-type)</td>
<td>See list</td>
<td>r3.8xlarge</td>
<td>String</td>
<td>Instance type, which must match the EC2 instance metadata string</td>
</tr>
<tr>
<td>cpu (CPUs)</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>Integer</td>
</tr>
<tr>
<td>core (Cores)</td>
<td>integer</td>
<td>16</td>
<td>Integer</td>
<td>Total number of processor cores</td>
</tr>
<tr>
<td>threads (threads per core)</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>Integer</td>
</tr>
<tr>
<td>t-ecu (total ECU value)</td>
<td>integer</td>
<td>32</td>
<td>Double</td>
<td>ECU value for previous-generation instance types that have ECU ratings; number of cores for post-ECU instance types</td>
</tr>
<tr>
<td>ecu (ECU per core)</td>
<td>double</td>
<td>1</td>
<td>Double</td>
<td>1 for all post-ECU instance types; total ECU divided by cores</td>
</tr>
<tr>
<td>hthread (hyperthreading)</td>
<td>thread</td>
<td>core</td>
<td>String</td>
<td>thread for hyperthreaded instance types; core for non-</td>
</tr>
</tbody>
</table>
User-configurable EBS Volume Types

The AWS Data Provider for SAP maintains a database of all relevant EBS volume types for SAP.

Entries for EBS volume types must be in a comma-separated list, as follows:

```
voltype,ebs-type,sample-time,
```

For example:

```
voltype,io1,60,
```

where the following applies:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Content</th>
<th>Example</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>keyword</td>
<td>voltype</td>
<td>—</td>
<td>String</td>
<td>A token to identify a record with an EBS volume description</td>
</tr>
<tr>
<td>ebs-type (EBS-type)</td>
<td>io1</td>
<td>gp2</td>
<td>sc1</td>
<td>st1</td>
</tr>
<tr>
<td>sample-time</td>
<td>60</td>
<td>300</td>
<td>60</td>
<td>Integer</td>
</tr>
</tbody>
</table>

**Important**

The sample time is required to calibrate the EBS metrics to the SAP monitoring requirements. Changes in the sample time will lead to incorrect EBS metrics in the SAP monitoring system.
User-Configurable Support Status

To include optional entries for the support status, use a comma-separated list, as follows:

```
support,status,
```

For example:

```
support,status,
```

where the following applies:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Content</th>
<th>Example</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>keyword</td>
<td>support</td>
<td>—</td>
<td>String</td>
<td>A token to identify the support status</td>
</tr>
<tr>
<td>status</td>
<td>production</td>
<td>anonymous</td>
<td>none</td>
<td>Choose <code>production</code> (default) to allow AWS to check whether the instance meets prerequisites</td>
</tr>
</tbody>
</table>

The `status` field supports the following values:

- **production** (default): Allows AWS to check whether the instance meets the SAP required prerequisites for SAP support, from SAP Note 1656250 (requires SAP support portal login).
- **anonymous**: Does not allow AWS to check whether the instance meets prerequisites.
- **none**: Legacy mode for version 1.3.1 backward compatibility.

Appendix B: Verification of AWS Data Provider for SAP in SAP System Monitoring

The AWS Data Provider for SAP exposes AWS-specific metrics through an XML page at `http://localhost:8888/vhostmd` of the given system.

This section explains which metrics get exposed to the SAP system and how you can access them for SAP system monitoring.

**Checking Metrics with the SAP Operating System Collector (SAPOSCOL)**

The information provided by the AWS Data Provider for SAP is read by the SAP Operating System Collector (SAPOSCOL). You can use the interactive mode of SAPOSCOL to verify that the two tools are working together correctly. The following example shows a lookup under Windows. A lookup under Linux is very similar.
1. Open a Windows command shell and direct the shell to the directory `C:\Program Files\SAP\hostctrl\exe`. Start `saposcol.exe` with the `-d` option.

![Figure 41: Starting SAPOSCOL](image1)

2. SAPOSCOL is now in interactive mode. Type `dump ccm` and press Enter to list all values gathered. SAPOSCOL will display a lengthy list of metrics, as shown here.

![Figure 42: Metrics from SAPOSCOL](image2)

The following two metrics indicate that SAPOSCOL is collaborating successfully with the AWS Data Provider for SAP:

- Enhanced Monitoring Access TRUE
- Enhanced Monitoring Details ACTIVE

The AWS-specific metrics start with the following strings:

- Virtualization_Configuration
- CPU_Virtualization_Virtual_System
- Memory_Virtualization_Virtual_System
- System_Info_Virtualization_System
SAPOSCOL hands the AWS-enhanced statistics with other operating system-specific metrics to the SAP system. You can also check the AWS-enhanced statistics in the SAP CCMS. You can enter the transaction st06 (or /nst06) in the upper-left transaction field of the SAP GUI for quick access to this data.

**Note**
You will need the appropriate authorizations to look up this information.
On this screen, you can verify core AWS information such as:

- Cloud provider
- Instance type
- Status of enhanced monitoring access (must be TRUE)
- Status of enhanced monitoring details (must be ACTIVE)
- Virtual machine identifier

**Important**
The enhanced AWS metrics aren't shown in standard view.

To view enhanced AWS statistics, choose the **Standard View** button in the upper-left corner. It changes to **Expert View** and displays the enhanced AWS statistics. The list that appears is comprehensive. It shows the processor details.

---

**Figure 44: Statistics in the SAP CCMS (standard view)**
It also shows details about the memory subsystem (main memory and disks) and network interfaces.

Figure 45: Enhanced AWS statistics (expert view)

Figure 46: Memory and networking statistics (expert view)
Note
The screen illustrations in Figures 37–39 were taken from SAP NetWeaver 7.4 SP08. This version shows the enhanced AWS statistics in the Memory Virtualization section. This problem has been fixed by SAP in later versions of NetWeaver.

Appendix C: Changes and Bug Fixes by Product Version

Version 1.2 (September 16, 2014)
- New: Support for the T2, R3, and C3 instance families.
- New: Support for post-ECU (EC2 Compute Unit) instance types:
  - New instance types no longer have ECU values.
  - The reference compute power for these instance types is a hardware thread of the given processor. The total CPU power is equal to the number of the vCPUs of a given instance type.
- New: Support for the new EBS GP2 volume type.
  - Every volume is now tagged with the EBS volume type.
  - EBS volumes now report their individual sample interval in a separate attribute.
- Bug fix: EBS volume mapping for Windows devices now reports the correct name.
- Bug fix: Installation, update, and operation through HTTP/HTTPS proxies has been fixed.
- New: JRE 8 support has been added on Linux.

Version 1.2.1 (September 29, 2014)
- Bug fix: EBS volumes now report correct attribute type ("string") for volume type.

Version 1.2.2 (October 1, 2014)
- Windows bug fix: Installer executable pulls installation from correct Amazon S3 bucket.
- Windows bug fix: AWS Data Provider for SAP now reports the correct disk mapping for Windows EBS volumes with the following names: xvd[a-z][a-z].

Version 1.3 (February 17, 2015)
- New: Support for new Amazon EC2 C4 instance family.
- Bug fix: Relative performance of c3.8xlarge instances is now reported correctly.
- New: CloudWatch and Amazon EC2 metrics access points:
  - Support for the EU (Frankfurt) Region was added.
  - Access points are user configurable. You can add information about new AWS Regions without having to install a new product version.
  - Access points are now updated from an internet-based database file. You can add new AWS Regions by updating a web-based configuration file and then restarting the daemon/service.
- New: Message log files with fixed disk space consumption are provided on Linux.
- New: User-configurable EC2 instance types are available.
- New: Web update support was added for future EC2 instance types without product updates.
• Bug fix: GP2 volumes now report the correct sample interval time.
• New: User-configurable sample times for new EBS volume types are now available.
• New: The AWS Data Provider for SAP now reports the virtualization type of the EC2 instance.

Version 1.3.1 (July 14, 2015)
• Bug fixes: Security fixes.
• New: Support for C4, D2, and M4 instance types. Users who migrate instances with installed 1.3 agents will automatically receive support for the new instance types through an updated configuration database on the web.

Version 2.0 (December 22, 2015)
• New: Windows devices in the range sdb to sdzz get correct SCSI device IDs assigned.
• New: Java VM consumption is now limited to 64 MB maximum heap size.

Version 2.1 (January 20, 2016)
• Support for Asia Pacific (Seoul) Region.
• Bug fix: Version 2.0 pulled files from an incorrect S3 bucket for installation. Version 2.0 needs to be uninstalled before version 2.1 is installed.

Version 2.5 (May 2, 2016)
• Bug fixes: Security and stability fixes in versions 2.2-2.4.
• New: Support for new Amazon EBS volume types:
  • Throughput Optimized HDD (st1)
  • Cold HDD (sc1)
• New: Support for the Amazon EC2 X1 instance family.

Version 2.6 (September 1, 2016)
• Bug fixes: Installation script checks for existence of wget
• Support for Oracle Linux.

Version 2.7 (December 21, 2016)
• Support for Canada (Central), US East (Ohio), and EU (London) Regions.
• Default access point resolution for common AWS Regions is added.

Version 2.8 (March 1, 2017)
• SLES 12, Red Hat 7, and Oracle Linux 7 will now use SYSTEMD to manage the daemon.
• Support for SLES and SLES for SAP 12 SP2.
• SLES 12 SP1 systems will get migrated from Linux services to SYSTEMD when trying to install the AWS Data Provider without having it de-installed first.
• Minor changes in logging texts.
• Support for R4 and M4 instance types.
• Updated Windows installation verification.
Version 2.9 (August 30, 2017)
- Added support for China Regions.
- Added Linux uninstaller.
- Linux installer can be customized to install from a custom S3 bucket.
- Silent installer for Windows (does not require any input).
- Improvements in determination of access points.
- Support for X1E instance family.

Version 3.0 (April, 2020)
- Initial release of the 3.0 version.
- Switched the Java Runtime from Oracle to Amazon Corretto.

Appendix D: Uninstalling older 2.X versions

Linux:
/usr/local/ec2/aws-agent/bin/aws-agent_uninstall

Windows:
"C:\Program Files\Amazon\DataProvider\uninstall.exe"

Appendix E: Example of Captured Metrics

This following show example metrics. Your system metrics may slightly differ.

<metrics>
<metric context="host" category="config" type="long" unit="posixtime">
  <name>Time Stamp</name>
  <value>1584376572</value>
</metric>
<metric context="host" category="config" type="int64" unit="sec">
  <name>Refresh Interval</name>
  <value>60</value>
</metric>
<metric context="vm" category="config" type="string" unit="none">
  <name>Data Provider Version</name>
  <value>3.0.139</value>
</metric>
<metric context="host" category="config" type="string" unit="none">
  <name>Cloud Provider</name>
  <value>Amazon Web Services</value>
</metric>
<metric context="vm" category="config" type="string" unit="none">
  <name>Instance Type</name>
  <value>m5.large</value>
</metric>
<metric context="host" category="config" type="string" unit="none">
  <name>Virtualization Solution</name>
  <value>KVM</value>
</metric>
</metrics>
<metric context="host" category="config" type="string" unit="none">
  <name>Virtualization Solution Version</name>
  <value>ba185a32</value>
</metric>

<metric context="host" category="config" type="long" unit="none">
  <name>CloudWatch Calls</name>
  <value>12</value>
</metric>

<metric context="host" category="config" type="long" unit="none">
  <name>EC2 Calls</name>
  <value>4</value>
</metric>

<metric context="vm" category="config" type="string" unit="none">
  <name>CPU Over-Provisioning</name>
  <value>no</value>
</metric>

<metric context="vm" category="config" type="string" unit="none">
  <name>Memory Over-Provisioning</name>
  <value>no</value>
</metric>

<metric context="vm" category="config" type="string" unit="none">
  <name>Virtualization Type</name>
  <value>default-hvm</value>
</metric>

<metric context="vm" category="config" type="long" unit="posixtime">
  <name>Last Hardware Change</name>
  <value>1572284007</value>
</metric>

<metric context="host" category="cpu" type="string">
  <name>Processor Type</name>
  <value>Intel(R) Xeon(R) @ 2500MHz</value>
</metric>

<metric context="host" category="cpu" type="int64">
  <name>Number of Cores per CPU</name>
  <value>1</value>
</metric>

<metric context="host" category="cpu" type="int64">
  <name>Number of Threads per Core</name>
  <value>2</value>
</metric>

<metric context="host" category="cpu" type="int64">
  <name>Max HW Frequency</name>
  <value>2500</value>
</metric>

<metric context="host" category="cpu" type="int64">
  <name>Current HW Frequency</name>
  <value>2500</value>
</metric>

<metric context="vm" category="cpu" type="string">
  <name>Reference Compute Unit (CU)</name>
  <value>Intel(R) Xeon(R) @ 2500MHz</value>
</metric>

<metric context="vm" category="cpu" type="int64">
  <name>vCPU Mappings</name>
  <value>thread</value>
</metric>

<metric context="vm" category="cpu" type="long">
  <name>Phys. Processing Power per vCPU</name>
  <value>1</value>
</metric>

<metric context="vm" category="cpu" type="int64">
  <name>Guaranteed VM Processing Power</name>
  <value>57</value>
</metric>
<value>2</value>
</metric>

<metric context="vm" category="cpu" type="int64" unit="cu">
<name>Current VM Processing Power</name>
<value>2</value>
</metric>

<metric context="vm" category="cpu" type="int64" unit="cu">
<name>Max. VM Processing Power</name>
<value>2</value>
</metric>

<metric context="vm" category="cpu" type="double" unit="percent">
<name>VM Processing Power Consumption</name>
<value>3.00</value>
</metric>

<metric context="vm" category="memory" type="long" unit="MB">
<name>Guaranteed Memory assigned</name>
<value>8274</value>
</metric>

<metric context="vm" category="memory" type="long" unit="MB">
<name>Current Memory assigned</name>
<value>8274</value>
</metric>

<metric context="vm" category="memory" type="long" unit="MB">
<name>Max Memory assigned</name>
<value>8274</value>
</metric>

<metric context="vm" category="memory" type="double" unit="percent">
<name>VM Memory Consumption</name>
<value>29.00</value>
</metric>

<metric context="vm" category="memory" type="int64" unit="KB/sec">
<name>Memory SwapIn Rate</name>
<value>0</value>
</metric>

<metric context="vm" category="memory" type="int64" unit="MB">
<name>Memory Swapped Out</name>
<value>0</value>
</metric>

<metric context="vm" category="memory" type="int64" unit="MB">
<name>Memory Lent</name>
<value>0</value>
</metric>

<metric context="vm" category="memory" type="int64" unit="KB/sec">
<name>Visible Memory SwapIn Rate</name>
<value>0</value>
</metric>

<metric context="vm" category="memory" type="int64" unit="MB">
<name>Visible Memory Swapped Out</name>
<value>0</value>
</metric>

<metric context="vm" category="network" type="int64" unit="bytes">
<name>Network Read Bytes</name>
<value>54110386</value>
</metric>

<metric context="vm" category="network" type="int64" unit="bytes">
<name>Network Write Bytes</name>
<value>1330726</value>
</metric>
Appendix E: Example of Captured Metrics

- **TCP Packets Retransmitted**: 396480
- **Minimum Network Bandwidth**: 10000 Mbps
- **Maximum Network Bandwidth**: 10000 Mbps
- **Mapping**: lan2
- **Volume Idle Time**: 58489 msec
- **Volume Queue Length**: 0
- **Volume Read Bytes**: 9878528 bytes
- **Volume Read Ops**: 144
- **Volume Read Time**: 246 msec
- **Volume Write Time**: 8266 msec
- **Volume Write Bytes**: 282332160 bytes
- **Volume Write Ops**: 3090
- **Volume Type**: gp2
Conclusion

This document helps SAP users to install and operate the AWS Data Provider for SAP. The AWS Data Provider for SAP allows SAP support to monitor SAP applications according to SAP standards.

Contributors

The following individuals contributed to this document:

- Sander Bleijenbergh, Solutions Architect, Amazon Web Services

Document Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>December 2020</td>
<td>Updated instructions for installation</td>
</tr>
<tr>
<td>October 2020</td>
<td>Windows installation instructions updated</td>
</tr>
<tr>
<td>June 2020</td>
<td>Minor updates to correct inaccurate content and links</td>
</tr>
<tr>
<td>April 2020</td>
<td>Update for software release 3.0</td>
</tr>
<tr>
<td>September 2017</td>
<td>Update for software release 2.9</td>
</tr>
<tr>
<td>December 2015</td>
<td>Update for software release 2.0</td>
</tr>
<tr>
<td>January 2014</td>
<td>Initial release</td>
</tr>
</tbody>
</table>
SAP on AWS Pricing and Optimization

*SAP Specialists, Amazon Web Services (AWS)*

*Last updated (p. 78): July 2019*

About this Guide

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the Amazon Web Services (AWS) Cloud. It explains how to estimate the cost of running your SAP environment on AWS.

For the other guides in the series, from overviews to advanced topics, see [SAP on AWS Technical Documentation](#).

Overview

For SAP customers and partners who are new to Amazon Web Services (AWS), the process of estimating your monthly AWS bill might seem a bit overwhelming at first. This guide explains how to estimate the cost of running your SAP environment on AWS.

The number and types of services offered by AWS has increased dramatically over time, but the AWS philosophy on pricing has not changed: at the end of each month, you pay only for what you use, and you can start or stop using a service at any time. No long-term contracts are required.

This guide is intended for SAP customers and partners who need to estimate the monthly cost of running SAP environments on AWS. For the purposes of this guide, we assume that you already know how to architect and size SAP solutions on AWS. If you do not, before you continue, we suggest that you read the [SAP on AWS Overview and Planning documentation](#).

To help you understand how to effectively estimate the cost of running your SAP environment on AWS, we provide three examples of SAP on AWS pricing using the AWS Simple Monthly Calculator. For each example, we review the architecture on AWS, example usage of each service, cost breakdown for each service, and total estimated monthly charge.

AWS Pricing Overview

AWS offers you a pay-as-you-go approach for pricing of more than 165 cloud services. With AWS you pay only for the individual services you use, for as long as you use them, and without requiring long-term contracts or complex licensing. AWS pricing is similar to how you pay for utilities, such as water and electricity. You only pay for the services you consume, and when you stop using them, there are no additional costs or termination fees.
For more information about AWS pricing, see How does AWS pricing work?

AWS Simple Monthly Calculator

The AWS Simple Monthly Calculator is an easy-to-use online tool that enables you to estimate the monthly cost of AWS services for your use case based on your expected usage. The AWS Simple Monthly Calculator is continuously updated with the latest pricing for all AWS services in all Regions. For an overview of how to use the AWS Simple Monthly Calculator, see the Getting Started with the AWS Simple Monthly Calculator video.

SAP on AWS Pricing Fundamentals

AWS currently offers over 165 different products and services. The following is an overview of the pricing characteristics for the AWS services that are most relevant for the deployment and operation of SAP systems on AWS.

AWS Region Pricing

AWS service pricing varies between different AWS Regions. The first step in estimating any SAP environment on AWS is to select which AWS Region you will deploy your SAP environment in.

Compute Pricing

Instances

Amazon Elastic Compute Cloud (Amazon EC2) provides a wide selection of instance types that provide varying combinations of CPU, memory, storage, I/O, and networking capabilities. You are charged by the hour for each running instance. The hourly fee of an instance is based on a combination of the following characteristics:

- **Instance type** – Specific virtual machine and bare metal configurations that offer different CPU, memory, storage, I/O, and networking capabilities. For more information about Amazon EC2 instance types, see Amazon EC2 Instance Types.

- **Operating system** – You can choose to buy an operating system license from AWS or bring your own operating system license or subscription. If you choose to buy the operating system license from AWS, the license fee is included in the EC2 instance fees. For more information, see SAP on AWS Overview and Planning.

- **Monitoring** – You can use Amazon CloudWatch to monitor your Amazon EC2 instances. Basic monitoring is included in the hourly cost of an instance and provides metrics at five-minute intervals. For an additional cost, you can use detailed monitoring, which provides metrics at one-minute intervals. Detailed monitoring is required for production SAP systems on AWS. For more information about Amazon CloudWatch pricing, see Amazon CloudWatch pricing.

- **I/O throughput** – Amazon EC2 instance types offer different levels of I/O throughput. For production SAP systems, either Amazon Elastic Block Store (Amazon EBS)-optimized instances or instances with 10 gigabit network connectivity are recommended. For more information about Amazon EBS-optimized instances and instances with 10 gigabit network connectivity, see Amazon EC2 Instance Configuration.

For more information about Amazon EC2 instance pricing, see Amazon EC2 pricing.
Purchasing Options

Amazon EC2 offers multiple purchasing options that give you flexibility to optimize your costs. The On-Demand, Reserved Instance, and Dedicated Host purchasing options are most appropriate for SAP systems.

On-Demand

With the On-Demand option, you pay for compute capacity by the hour with no long-term commitments or upfront payments. You can increase or decrease your compute capacity depending on the demands of your application, and pay only the specified hourly rate for the instances you use.

Recommended for:
- Initial purchasing option for SAP on AWS projects
- Temporary SAP systems for testing, upgrades, trials, demos, and proof of concepts (POCs)
- Temporary SAP systems for scaling to handle increased load

Reserved Instances

The Reserved Instance purchasing option provides you with a significant discount (up to 75%) compared to On-Demand instance pricing. In addition, when Reserved Instances are assigned to a specific Availability Zone, they provide a capacity reservation, which enables you to launch instances when you need them.

For applications that have steady state or predictable usage, Reserved Instances can provide significant savings compared to On-Demand Instances. For more information, see How to Purchase Reserved Instances.

Recommended for:
- Permanent SAP systems such as production, quality assurance, and development systems
- Consider for any SAP system that will be online more than 60% of the time

Dedicated Hosts

A Dedicated Host is a physical Amazon EC2 server dedicated for your use. Dedicated Hosts enable you to use your existing server-bound software licenses, including Windows Server and SQL Server.

**Note**

Dedicated Hosts are required for Amazon EC2 High Memory instances (6 TB, 9 TB, and 12 TB of memory) and Amazon EC2 bare metal instances.

For more information about Amazon EC2 purchasing options, see Amazon EC2 Pricing.

Storage Pricing

AWS provides flexible, cost-effective, and easy-to-use data storage options for your SAP systems. Each option has a unique combination of performance and durability. This section provides an overview of the primary components of AWS storage pricing.

Primary Storage

Amazon Elastic Block Store (Amazon EBS) provides persistent, block-level storage volumes for SAP systems that run on Amazon EC2. Each SAP Amazon EC2 system requires one or more Amazon EBS
volumes to store system components, such as the operating system, SAP software, SAP database data and log files, and local backup storage. For information about Amazon EBS pricing, see Amazon EBS Pricing.

Backup Storage

Multiple options are available to back up your SAP system on AWS. For an overview of options, see SAP on AWS Backup and Recovery Guide. Each option stores backup files in Amazon Simple Storage Service (Amazon S3) to provide highly durable storage that is independent of primary storage. For information about Amazon S3 pricing, see Amazon S3 Pricing.

The amount of backup storage required depends on the backup method, backup frequency, system size, and backup retention. How to calculate the amount of backup storage you require is not included in this guide.

Network Pricing

Amazon VPC Connectivity

There is no additional cost for using Amazon Virtual Private Cloud (Amazon VPC) over the standard Amazon EC2 usage charges. If a secure connection is required between your on-premises network and Amazon VPC, you can choose one of the following connectivity options.

Hardware VPN Connection

When you use hardware VPN connections to your Amazon VPC, you are charged for each VPN Connection-hour that your VPN connection is provisioned and available. For more information about hardware VPN connection pricing, see Amazon VPC pricing.

Private Network Connection

With AWS Direct Connect, you can make a dedicated network connection from your on-premises network to AWS. AWS Direct Connect is billed by port hours and outbound data transfer rate. For more information about AWS Direct Connect pricing, see AWS Direct Connect pricing.

Data Transfer

There is no charge for inbound data transfer to Amazon EC2. Charges do apply for data that is transferred out from Amazon EC2 to the internet, to another AWS Region, or to another Availability Zone. For details on AWS data transfer pricing, see the On-Demand section of the Amazon EC2 Pricing page.

AWS Support Pricing

AWS offers different levels of support, so you can choose the right level of support for your environment. AWS Basic support is included with all AWS services at no additional cost. If you require a deeper level of support, you can subscribe to Developer, Business, or Enterprise-level support. For more information about the different AWS support plans and pricing details, see AWS Support.

SAP on AWS Pricing Examples

To help you estimate the monthly cost of your SAP environment on AWS, review the following sample SAP on AWS environment estimates created using the AWS Simple Monthly Calculator.
• the section called “SAP HANA – Multi-AZ (HA) Single-Node Architecture – Up to 4 TB Memory” (p. 65)
• the section called “SAP HANA – Multi-AZ (HA) – Single-Node Architecture – 6–12 TB Memory” (p. 69)
• the section called “SAP S/4HANA – Multi-AZ (HA) – 3-Tier Architecture” (p. 74)

SAP HANA – Multi-AZ (HA) Single-Node Architecture – Up to 4 TB Memory

Description

The architecture in this example is based on the *SAP HANA Multi-AZ (HA), single-node architecture* described in the *SAP HANA Quick Start* guide. This architecture provisions two Amazon EC2 instances in private subnets in two different Availability Zones. High availability (HA) is based on SLES High Availability Extension (HAE), which is part of the SLES for SAP operating system. The SLES for SAP operating system uses a bring-your-own-subscription, or you can purchase a subscription through the AWS Marketplace. SAP HANA System Replication (HSR) synchronous replication is replicates between the primary node and secondary node. The SAP HANA nodes are run on the same sized Amazon EC2 instances types, and each system has its own set of Amazon EBS volumes.

- SAP HANA single node or scale-up system with 2 TB of memory
- SAP HANA scenarios – Include data mart, analytics, native SAP HANA application, ERP on HANA, S/4HANA, and BW/4HANA
- SAP HANA systems run in a private subnet and are not directly accessible from the internet
- Administrative and end user system access through a Windows Remote Desktop Services system running within a public subnet, which is accessible through a direct internet connection (VPN connection not required)

The following are sample compute requirements mapped to relevant Amazon EC2 instance types.

<table>
<thead>
<tr>
<th>System</th>
<th>SAPS</th>
<th>Memory</th>
<th>Amazon EC2 Instance Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP HANA DB – Primary</td>
<td>130,000</td>
<td>2 TB</td>
<td>x1.32xlarge</td>
</tr>
<tr>
<td>SAP HANA DB – Secondary</td>
<td>130,000</td>
<td>2 TB</td>
<td>x1.32xlarge</td>
</tr>
<tr>
<td>RDP</td>
<td>N/A</td>
<td>6 GB</td>
<td>c4.xlarge</td>
</tr>
<tr>
<td>Bastion host</td>
<td>N/A</td>
<td>2 GB</td>
<td>t2.small</td>
</tr>
</tbody>
</table>

Architecture

The following diagram shows the SAP HANA – Multi-AZ (HA) – Single-node architecture used in this example.
Figure 1 – SAP HANA – Multi-AZ (HA) – Single-node architecture

AWS Simple Monthly Calculator

You can see this sample estimate online in the AWS Simple Monthly Calculator at: SAP HANA – Multi-AZ (HA) – Single-Node Architecture – Up to 4 TB Memory

The following images show each part of the AWS Simple Monthly Calculator estimate.

Figure 2 – Amazon EC2 Instances example settings

Note
Because this sample architecture uses SLES for the SAP operating system and a bring-your-own-subscription model on AWS, it uses a Linux operating system. If you do not require SLES for SAP and want to use the bundled standard SLES software and license, you select a SUSE Linux Enterprise Server for the operating system. For more information, see the SAP on AWS Overview and Planning guide.

1. Bastion host system
2. RDP system
3. HANA Primary system
4. HANA Secondary system
5. All instances are set at 100% utilization
6. Amazon EC2 instance type selected
7. Linux operating system
8. Linux operating system
9. Amazon EC2 purchasing option – Reserved Instance 3 Year No Upfront Payment

Figure 3 – Amazon EBS Volumes example settings

A & B – HANA primary and secondary nodes are deployed in different Availability Zones and each have their own set of EBS volumes.

1. Bastion root volume
2. RDP C:\ drive volume
3. HANA root volume
4. HANA /usr/sap volume
5. HANA metadata volume
6. HANA Shared volume
7. HANA Data volume
8. HANA Log volume
9. HANA Backup volume
10. EBS volume type selected for each volume
11. Native or third-party backup tools are used to perform system backups.
12. Amazon EBS snapshots are not used, so there is no data in the Snapshot Storage fields.
Figure 4 – Elastic IP Address, Data Transfer, and Elastic Load Balancing example settings

1. One Elastic IP address per running instance is provided at no charge. Additional Elastic IP addresses are not required.
2. 2 TB per month is estimated for data transferred out to the internet.
3. 2 TB per month is estimated for data transferred in from the internet.
4. 500 GB per month Intra-Region Data Transfer for database replication between HANA Primary system in Availability Zone 1 and HANA Secondary system in Availability Zone 2.

Figure 5 – Amazon S3 example settings

1. 4 TB of Amazon S3 storage is estimated for storing system and database backups. The actual backup storage required depends on the backup method, frequency, volume, and retention policy.
2. 1,000 requests are estimated for Amazon S3 PUT, COPY, POST, and LIST operations, and 1,000 requests are estimated for GET and other operations.
SAP HANA – Multi-AZ (HA) – Single-Node Architecture – 6–12 TB Memory

Description

The architecture in this example is based on the SAP HANA Multi-AZ (HA), single-node architecture described in the SAP HANA Quick Start guide. This architecture provisions two Amazon EC2 instances in private subnets in two different Availability Zones. High availability is based on SLES High Availability Extension (HAE), which is part of the SLES for SAP operating system. The SLES for SAP operating system uses a bring-your-own-subscription model, or you can purchase a subscription through the AWS Marketplace. SAP HANA System Replication (HSR) synchronous replication is used to replicate between the primary node and secondary node. The SAP HANA nodes run on the same-sized Amazon EC2 instances types and each system has its own set of Amazon EBS volumes.

- SAP HANA scenarios – Include data mart, analytics, native SAP HANA application, ERP on HANA, S/4HANA, and BW/4HANA
- SAP HANA single node and scale-up system with 9 TB of memory
- High Availability
- SAP HANA system in a private subnet and not directly accessible from the internet
- Administrative and end user system access through a Windows Remote Desktop Services system running within a public subnet, which is accessible through a direct internet connection (VPN connection not required)
The following are sample compute requirements mapped to the relevant Amazon EC2 instance types.

<table>
<thead>
<tr>
<th>System</th>
<th>SAPS</th>
<th>Memory</th>
<th>Amazon EC2 Instance Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP HANA DB – Primary</td>
<td>130,000</td>
<td>9 TB</td>
<td>u-9tb1.metal</td>
</tr>
<tr>
<td>SAP HANA DB – Secondary</td>
<td>130,000</td>
<td>9 TB</td>
<td>u-9tb1.metal</td>
</tr>
<tr>
<td>RDP</td>
<td>N/A</td>
<td>6 GB</td>
<td>c4.xlarge</td>
</tr>
<tr>
<td>Bastion host</td>
<td>N/A</td>
<td>2 GB</td>
<td>t2.small</td>
</tr>
</tbody>
</table>

### Architecture

The following diagram shows the SAP HANA – Multi-AZ (HA) – Single-Node Architecture in this example.

![Figure 7 – SAP HANA – Multi-AZ (HA) – Single Node Architecture](image)

### AWS Simple Monthly Calculator

You can see this sample estimate online in the AWS Simple Monthly Calculator at: [SAP HANA – Multi-AZ (HA) Single-Node Architecture – 6 TB and larger](Link)

The following images show each part of the AWS Simple Monthly Calculator estimate.
Figure 8 – Amazon EC2 example settings

A. On-demand Amazon EC2 instances
B. Amazon EC2 High Memory and bare metal instance types require Amazon EC2 Dedicated Hosts

1. Bastion host – EC2 instance
2. RDP – EC2 instance
3. All instances are set at 100% utilization
4. Amazon EC2 purchasing option
5. HANA Primary system
6. HANA Secondary system
7. All instances are set at 100% utilization
8. Amazon EC2 purchasing option
Figure 9 – Amazon EBS Volumes example settings

A & B – The HANA primary and secondary nodes are deployed in different Availability Zones and each have their own set of EBS volumes.

1. Bastion host root volume
2. RDP server C drive volume
3. HANA root volume
4. HANA /usr/sap volume
5. HANA metadata volume
6. HANA Shared volume
7. HANA Data volumes
8. HANA Log volume
9. HANA Backup volume

3. EBS volume type
4. Native or third-party backup tools are used to perform system backups. Amazon EBS snapshots are not used, so there is no data in the Snapshot Storage fields.

Figure 10 – Elastic IP Address, Data Transfer, and Elastic Load Balancing example settings

1. One Elastic IP address per running instance is provided at no charge. Additional Elastic IP addresses are not required.
2. 2 TB per month is estimated for data transferred out to the internet.
3. 2 TB per month is estimated for data transferred in from the internet.
4. 500 GB per month Intra-Region Data Transfer for database replication between HANA Primary system in Availability Zone 1 and HANA Secondary system in Availability Zone 2.
Figure 11 – Amazon S3 example settings

1. 4 TB of Amazon S3 storage is estimated for storing system and database backups. The actual backup storage required depends on the backup method, frequency, volume, and retention policy.

2. 1,000 requests are estimated for Amazon S3 PUT, COPY, POST, and LIST operations, and 1,000 requests are estimated for GET and other operations.

Figure 12 – Estimate of monthly bill example

1. Total and detailed breakdown of Amazon EC2 costs
2. Total and detailed breakdown of Amazon S3 costs
3. Total and detailed breakdown of data transfer costs
4. Total and detailed breakdown of AWS Support costs
5. AWS Free Tier discount
6. Total monthly charge for all AWS services

**SAP S/4HANA – Multi-AZ (HA) – 3-Tier Architecture**

**Description**

The architecture in this example is for an SAP S/4HANA 3-tier (PRD/QAS/DEV) environment with Multi-AZ high availability for production.

The following are sample compute requirements mapped to the relevant Amazon EC2 instance types.

<table>
<thead>
<tr>
<th>System</th>
<th>Landscape</th>
<th>SAPS</th>
<th>Memory</th>
<th>Amazon EC2 Instance Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP S/4HANA DB</td>
<td>PRD</td>
<td>130,000</td>
<td>2 TB</td>
<td>x1.32xlarge</td>
</tr>
<tr>
<td>SAP S/4HANA NW AS</td>
<td>PRD</td>
<td>10,000</td>
<td>64 GB</td>
<td>r5.2xlarge</td>
</tr>
<tr>
<td>SAP S/4HANA DB</td>
<td>PRD - HA</td>
<td>130,000</td>
<td>2 TB</td>
<td>x1.32xlarge</td>
</tr>
<tr>
<td>SAP S/4HANA NW AS</td>
<td>PRD - HA</td>
<td>10,000</td>
<td>64 GB</td>
<td>r5.2xlarge</td>
</tr>
<tr>
<td>SAP S/4HANA DB</td>
<td>QAS</td>
<td>130,000</td>
<td>2 TB</td>
<td>x1.32xlarge</td>
</tr>
<tr>
<td>SAP S/4HANA NW AS</td>
<td>QAS</td>
<td>10,000</td>
<td>64 GB</td>
<td>r5.2xlarge</td>
</tr>
<tr>
<td>SAP S/4HANA DB</td>
<td>DEV</td>
<td>5,000</td>
<td>32 GB</td>
<td>x1.16xlarge</td>
</tr>
<tr>
<td>SAP S/4HANA NW AS</td>
<td>DEV</td>
<td>5,000</td>
<td>32 GB</td>
<td>r5.xlarge</td>
</tr>
<tr>
<td>SAP SolMan</td>
<td>N/A</td>
<td>10,000</td>
<td>16 GB</td>
<td>c5.2xlarge</td>
</tr>
<tr>
<td>SAP SAProuter</td>
<td>N/A</td>
<td>N/A</td>
<td>2 GB</td>
<td>t2.small</td>
</tr>
</tbody>
</table>

**Architecture**

The following diagram shows the SAP S/4HANA – Multi-AZ (HA) – 3 Tier architecture in this example.
Figure 13 – SAP S/4HANA – Multi-AZ (HA) – 3-Tier Architecture

AWS Simple Monthly Calculator

This sample estimate is available online in the AWS Simple Monthly Calculator at SAP S/4HANA – Multi-AZ (HA) – 3-tier (PRD/QAS/DEV) architecture.

The following images show each part of the AWS Simple Monthly Calculator estimate.

Figure 14 – Amazon EC2 Instances example settings

1. PRD – SAP HANA DB system
2. PRD – SAP NetWeaver AS system
3. PRD-HA – SAP HANA DB system
4. PRD HA – SAP NetWeaver AS system
5. QAS – SAP HANA DB system
6. QAS – SAP NetWeaver AS system
7. DEV – SAP HANA DB system
8. DEV – SAP HANA NetWeaver AS system
9. SAP Solution Manager system
10. SAProuter system
11. Usage 100%
12. Amazon EC2 instance type selected for each system
13. Amazon EC2 purchasing option selected for each system

Figure 15 – Amazon EBS Volumes example settings

Note
To simplify the example, Amazon EBS requirements for each SAP HANA system have been aggregated into a single line item. For detailed Amazon EBS volume configuration information see the SAP HANA Quick Start guide.

1. PRD – HANA DB volume - General Purpose SSD (gp2)
2. PRD – NetWeaver AS volume - General Purpose SSD (gp2)
3. PRD-HA – HANA DB volume - General Purpose SSD (gp2)
4. PRD-HA – NetWeaver AS volume - General Purpose SSD (gp2)
5. QAS – HANA DB volume - General Purpose SSD (gp2)
6. QAS – NetWeaver AS volume - General Purpose SSD (gp2)
7. DEV – HANA DB volume - General Purpose SSD (gp2)
8. DEV – NetWeaver AS volume – General Purpose SSD (gp2)
9. Solution Manager volume – General Purpose SSD (gp2)
10. SAProuter volume – General Purpose SSD (gp2)
11. EBS volume type
12. Total amount of storage in GB estimated per system
13. Native or third-party backup tools are used to perform system backups. Amazon EBS snapshots are not used, so there is no data in the Snapshot Storage fields.
Figure 16 – Elastic IP Address, Data Transfer, and Elastic Load Balancing example settings

1. One Elastic IP address per running instance is provided at no charge. Additional Elastic IP addresses are not required.
2. 2 TB per month is estimated for data transferred out to the internet.
3. 2 TB per month is estimated for data transferred in from the internet.
4. 500 GB per month Intra-Region Data Transfer for database replication between HANA primary system in Availability Zone 1 and HANA secondary system in Availability Zone 2.

Figure 17 – Amazon S3 example settings

1. 16 TB of Amazon S3 storage is estimated for storing system and database backups. The actual backup storage required depends on the backup method, frequency, volume, and retention policy.
2. 1,000 requests are estimated for Amazon S3 PUT, COPY, POST, and LIST operations, and 1,000 requests are estimated for GET and other operations.
Figure 18 – Estimate of monthly bill example

1. Total and detailed breakdown of Amazon EC2 costs
2. Total and detailed breakdown of Amazon S3 costs
3. Total and detailed breakdown of data transfer costs
4. Total and detailed breakdown of AWS Support costs
5. AWS Free Tier discount
6. Total monthly charge for all AWS services

<table>
<thead>
<tr>
<th>Date</th>
<th>Change</th>
<th>In sections</th>
</tr>
</thead>
<tbody>
<tr>
<td>July 2019</td>
<td>Initial publication</td>
<td>—</td>
</tr>
</tbody>
</table>
Architecture Guidance for Availability and Reliability of SAP on AWS

Overview

This guide provides a set of architecture guidelines, strategies, and decisions for SAP customers and partners who have a requirement for deploying SAP NetWeaver-based systems with a highly available and reliable configuration on AWS.

In this guide we will cover:

- An Introduction to SAP high availability and reliability
- The Architecture Guidelines and Decisions to consider
- A set of Architecture Patterns and advice on when to use each pattern

This guide is intended for users who have previous experience designing High Availability and Disaster Recovery Architectures for SAP.

We do not cover the business requirements that determine the need for High Availability and Disaster Recovery, but we assume that you have completed the process to define and map your business requirements to your availability and reliability requirements.

This document does not cover the implementation details on a specific partner or custom solution. You will still need to assess and map capabilities of your implementation and solution to match your requirements.

Prerequisites

Specialized Knowledge

Before you follow the configuration instructions in this guide, we recommend that you become familiar with the following AWS services. (If you are new to AWS, see Getting Started with AWS.)
Recommended Reading

Before reading this document, we recommend that you read the Reliability pillar in the AWS Well-Architected Framework document to gain foundational knowledge of key concepts for architecting solutions on AWS.

We also recommend you first read some key overview and best practice guides:

- SAP on AWS Overview and Planning
- Getting Started with Architecting SAP on the AWS Cloud

Introduction

For decades, SAP customers have protected their mission critical SAP workloads on premise with two common patterns, High Availability and Disaster Recovery.

The advent of cloud computing provides an opportunity to rethink the approach to providing high availability and disaster recovery capabilities for SAP and move to thinking more about architecting for high availability and reliability, using modern architectures and technologies that are not possible in traditional data centers.

Before we go into further detail on how to approach this topic with a new way of thinking, it’s important to recap on the SAP system design and the single points of failure that are inherently a part of the SAP n-tier architecture.

SAP NetWeaver Architecture Single Points of Failure

[Diagram showing SAP NetWeaver Architecture with single points of failure]

- Web Dispatcher
- SAP’s Applications run on the Application Servers
- SAP Central Services (SCS) run critical components (Enqueue and Message Server)
- Shared NFS File Systems across all components
- Primary Application Server
- Additional Application Server
- SAP Central Services
- Database Instance
- SAP Application code and user data reside in the Single Database instance.

Web dispatcher load balances HTTP(S) connections across application servers based on metrics from application servers.
Figure 1: SAP single points of failure

Figure 1 shows the typical SAP NetWeaver architecture, which has several single points of failure which are listed below:

- SAP Central Services (Message Server and Enqueue Processes)
- SAP Application Server
- NFS (Shared Storage)
- Database
- SAP Web Dispatcher

For the SAP Central Services and Database, it is possible to deploy additional hosts to add protection for these single points of failure. For example, an additional host running the SAP replicated Enqueue to protect the loss of application level locks (Enqueue Locks) and an additional host running a secondary database instance to protect against data loss.

However, the inherent design of these single points of failure limit the ability to easily take advantage of cloud native features to provide high availability and reliability.

An example of a cloud native approach would be the use of a shared state solution designed to run across multiple physical locations (AWS Availability Zones) which would prevent the loss of the state held in the SAP Central Services component in the event of a hardware or software failure.

Another example of where a cloud native service can help protect one of the SAP single points of failure is Amazon Elastic File System (Amazon EFS) which is a highly available and durable managed NFS service which runs actively across multiple physical locations (AWS Availability Zones).

High Availability and Disaster Recovery

High Availability (HA) is the attribute of a system to provide service during defined periods, at acceptable or agreed upon levels and to mask unplanned outages from end-users. This is often achieved by using clustered servers providing automated failure detection and recovery or highly resilient hardware, as well as robust Testing, and Problem and Change Management.

Disaster Recovery (DR) protects against unplanned major outages such as site disasters through reliable, predictable recovery, usually on different hardware and physical location. The loss of data as a result of corruption, malware, and etc. is considered as a logical disaster event, and is normally technically resolved in a separate solution such as recovery from the latest backup or storage snapshot. Logical DR does not necessarily imply the failover to another facility.

From the perspective of documented and measurable data points, High Availability and Disaster Recovery requirements are often defined in terms of % Uptime, MTTR, RTS, RTO, and RPO.

- **Percentage Uptime**: Defines percentage of uptime in a given period (normally a month or a year)
- **Mean Time to Recovery (MTTR)**: The average time to recover from failure
- **Return to Service (RTS)**: The time it takes to bring the system back to service for the users
- **Recovery Time Objective (RTO)**: Defines the maximum amount of time a system or service can be down, how long a solution takes to recover, and the time it takes for a service to be available again.
- **Recovery Point Objective (RPO)**: Defines how much data a business is willing to lose, expressed in time. It's the maximum time between a failure and the recovery point.
On-Premises vs. Cloud deployment patterns

Traditionally, customers with High Availability requirements would deploy their primary compute capabilities in a single datacenter or hosting facility, often in two separate rooms or data center halls with disparate cooling and power, and high-speed network connectivity between them. Other customers would run two hosting facilities in close proximity, with a separation of compute capabilities, yet close enough to not be impacted by network latency.

To meet disaster recovery requirements and because these scenarios represent an elevated risk to unforeseen location failure, many customers would extend their architecture to include a secondary location where a copy of their data resided, often with additional idle compute capacity. The distance between the primary and secondary locations often created the need for asynchronous transfer of data which impacted the RPO that could be achieved. This was the standard and generally accepted architecture pattern for HA and DR for many industries and companies running SAP.

Customers migrating their SAP workloads to cloud providers today will in many cases still revert to this architecture and map it to AWS Regions and Availability Zones (AZ) as depicted in Figure 4 below. While this architecture can work in your environment, the architecture does not follow the AWS Well-Architected Framework which was developed by AWS to help cloud architects build secure, high-performing, resilient, and efficient infrastructure for their applications.
The approach AWS takes in geographically isolating facilities in Regions and AZs gives you an opportunity to and reconsider the fundamentals which have governed the standard architecture pattern. The multiple AZ approach provides distance while maintaining performance for the primary compute capacity, and greatly reduces the risk of location failure. This approach is shown in Figure 5 below.

With the risk of location failure significantly reduced for the primary compute capacity, the requirements for a second Region can be evaluated further depending on your business requirements. Idle hardware is no longer an issue, since with AWS you can rapidly deploy the required capacity, in the same or different Region. Data backups can be stored on Amazon Simple Storage Service (Amazon S3) in a single AWS Region or in multiple AWS Regions by leveraging cross-Region replication. This architecture can be simplified and be made readily available, while reducing the cost as shown in Figure 6 below.

In addition to considering the impact of infrastructure or hosting facility failure, another scenario to consider is the loss of business data due to accidental or malicious technical activity.

This scenario excludes complete physical loss of the business data which would fall under the scenarios involving loss of physical hosting locations. This is often referred to as logical DR, which requires a decision to be made to restore the business data from a local good copy of the data. To enable this, decisions have to be made as to where to store this business data and how it will be used in the event of a logical DR.

In the rest of this document we detail the key architecture guidelines and decisions you need to consider along with the availability architecture patterns to select to meet your specific availability and reliability requirements.
Architecture Guidelines and Decisions

This section will provide a brief overview of the AWS services typically used for SAP workloads and some of the key points to understand when designing your architecture for hosting SAP on AWS. If you are already familiar with these AWS services then you can skip this section.

Regions and Availability Zones

The AWS Global Infrastructure consists of AWS Regions Availability Zones (AZ). More details on the AWS Global Infrastructure can be found here.

Regions

To support its global footprint and ensure customers are served across the world, AWS maintains multiple geographic Regions, including Regions in North America, South America, Europe, Asia Pacific, and the Middle East.

An AWS Region is a collection of AWS resources in a geographic area. Each AWS Region is isolated and independent of the other Regions. For a list of Region names and codes, see Regional endpoints.

Regions provide fault tolerance, stability, and resilience. They enable you to create redundant resources that remain available and unaffected in the unlikely event of a Regional outage.

AWS Regions consist of multiple (typically 3) Availability Zones (AZ), each of which is a fully isolated partition of the AWS infrastructure that consists of discrete data centers, each with redundant power, networking, and connectivity, and each housed in separate facilities.

You retain complete control and ownership over the AWS Region in which your data is physically located, making it easy to meet Regional compliance and data residency requirements.

Availability Zones

Availability Zones (AZ) enable customers to operate production applications and databases that are more highly available than would be possible from a single data center. Distributing your applications across multiple availability zones provides the ability to remain resilient in the face of most failure modes, including natural disasters or system failures.

Each AZ can be multiple data centers, and at full scale can contain hundreds of thousands of servers. They are fully isolated partitions of the AWS global infrastructure. With their own powerful infrastructure, the AZs are physically separated by a meaningful distance, many kilometers, from any other AZ, although all are within 100 km (60 miles of each other) which provides isolation from the most common disasters that could affect data centers (i.e. floods, fire, severe storms, earthquakes, etc.).

All AZs within a Region are interconnected with high-bandwidth, low-latency networking, over fully redundant, dedicated metro fiber providing high-throughput, low-latency networking between AZs. The network performance is sufficient to accomplish synchronous replication between AZs.

When deciding on a location of an AZ, we do a deep analysis (for example, we review 100-year flood data) of all of potential sites and use this information to decide on placement. Ultimately, we evaluate placement of data centers based on the maximum distance defined by acceptable latency because our customers want to run their applications in a very highly available manner. To be highly available, an application needs to run in more than one location at a time with exactly the same data in each, thus allowing for a seamless failover with minimal downtime if a disaster should occur.

Services

Our general policy is to deliver AWS services, features, and instance types to all AWS Regions within 12 months of general availability, based on a variety of factors such as customer demand, latency, data
sovereignty, and other factors. You can share your interest for local Region delivery, request service roadmap information, or gain insight on service interdependency (under NDA) by contacting your AWS sales representative.

Due to the nature of the service, some AWS services are delivered globally rather than Regionally, such as Amazon Route 53, Amazon Chime, Amazon WorkDocs, Amazon WorkMail, Amazon WorkSpaces, and Amazon WorkLink.

Other services such as Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Elastic Block Store (EBS) are zonal services, and when you create an Amazon EC2 or EBS resource you are required to specify the required AZ within a Region to launch that resource in.

### Selecting the AWS Region(s)

When choosing which AWS Region(s) to deploy your SAP environment in, you should consider the following topics:

- Proximity to your on-premises data center(s), systems, and end users to minimize network latency.
- Data residency and compliance requirements.
- Whether the AWS products and services you plan to use are available in the Region. For a detailed list of AWS products and services by Region, see Region Table.
- The Amazon EC2 instances types you plan to use are available in the Region. To view AWS Region availability for a specific instance type, see the Amazon EC2 Instance Types for SAP page.
- AWS service pricing varies between different AWS Regions. When estimating any SAP environment on AWS you must first select which AWS Region you will deploy your SAP environment in. For more details, see the SAP on AWS Pricing and Optimization guide.

### Multi-Region Considerations

When deploying across multiple Regions, an important consideration is the associated cost and management effort for core services required in each Region such as networking, security, and audit services.

#### Network Latency

If you decide on a multiple Region approach, you should consider the impact of any increase in the network latency to the secondary Region from your on-premises locations.

#### Cross Regional Data Transfer

AWS provides several methods of data transfer between Regions which are relevant when designing an SAP Architecture for Disaster Recovery. You should also consider any data residency requirements when transferring data to another AWS Region as well as the costs associated with the data transfer (cross Region peering (p. 90) and/or S3 replication (p. 91)) and storage in the secondary Region.

#### Tier 0 Services

When using an AWS Region, there are a number of Tier 0 services that you will require to be available before you can deploy an SAP workload. These include DNS, Active Directory, and/or LDAP as well as any AWS- or ISV-provided security and compliance products and services.

### AWS Accounts

While there is no one-size-fits-all answer for how many AWS accounts a particular customer should have, most organizations will want to create more than one AWS account, because multiple accounts provide the highest level of resource and billing isolation.
General SAP Guides

In the context of SAP workloads, it is common for customers to deploy the Production environment in a separate AWS account for better isolation from the rest of the SAP landscape.

AWS Organizations is an account management service that enables you to consolidate multiple AWS accounts into an organization that you create and centrally manage. AWS Organizations includes account management and consolidated billing capabilities that enable you to better meet the budgetary, security, and compliance needs of your business. As an administrator of an organization, you can create accounts in your organization and invite existing accounts to join the organization.

AWS Landing Zone is a solution that helps customers more quickly set up a secure, multi-account AWS environment based on AWS best practices. This solution can help save time by automating the setup of an environment for running secure and scalable workloads while implementing an initial security baseline through the creation of core accounts and resources. It also provides a baseline environment to get started with a multi-account architecture, identity and access management, governance, data security, network design, and logging.

Note: The AWS Landing Zone solution is delivered by AWS Solutions Architects or Professional Services consultants to create a customized baseline of AWS accounts, networks, and security policies.

Consider using the AWS Landing Zone solution if you are looking to set up a configurable landing zone with rich customization options through custom add-ons such as, Active Directory, and change management through a code deployment and configuration pipeline.

AWS Control Tower provides the easiest way to set up and govern a secure, compliant, multi-account AWS environment based on best practices established by working with thousands of enterprises. With AWS Control Tower, your distributed teams can provision new AWS accounts quickly. Meanwhile your central cloud administrators will know that all accounts are aligned with centrally established, company-wide compliance policies.

Consider using AWS Control Tower if you are looking for a self-service experience to set up a new AWS environment based on a landing zone with pre-configured blueprints and then interactively govern your accounts with pre-configured guardrails.

Compute

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud. An Amazon EC2 instance is launched in a specific AZ within a specified Amazon Virtual Private Cloud (Amazon VPC).

When the EC2 instances are deployed across two or more AZs within a single Region then AWS offers an SLA of 99.99%.

Instance Types

A range of Amazon EC2 instance types are supported by SAP. When selecting the instance types for your SAP workload you should consider which tiers allow flexibility on the instances used (such as the application tier) and which tiers (such as the database tier) will require the use of a specific instance type based on compute, memory, storage throughput, and license compliance requirements.

For the tiers that you have specific instance type requirements without flexibility to change during a failure scenario, you should consider having a capacity reservation using Zonal Reserved Instances or on-demand capacity reservations within the required AZs and Regions where the instance will run. This approach is called Static Stability. For more information, see Static stability using Availability Zones.

Reserved Instances

Reserved Instances (RI) provide you with significant savings on your Amazon EC2 costs compared to On-Demand Instance pricing. Reserved Instances are not physical instances, but rather a billing discount
applied to the use of On-Demand Instances in your account. These On-Demand Instances must match certain attributes, such as instance type and Region, in order to benefit from the billing discount.

When deploying Amazon EC2 across multiple AZs for high availability, it is recommended that you consider using Zonal Reserved Instances. In addition to the savings over the on-demand instance pricing, a zonal Reserved Instance provides a capacity reservation in the specified AZ. This is to ensure that the capacity is available at the time you need it.

For billing purposes, the consolidated billing feature of AWS Organizations treats all the accounts in the organization as one account. This means that all accounts in the organization can receive the hourly cost benefit of Reserved Instances that are purchased by any other account.

**Savings plans**

Savings Plans are a flexible pricing model that provides savings of up to 72% on your AWS compute usage. This pricing model offers lower prices on Amazon EC2 instances usage, regardless of instance family, size, tenancy or AWS Region, and also applies to AWS Fargate and AWS Lambda usage.

Savings Plans offer significant savings over On-Demand, just like EC2 Reserved Instances, in exchange for a commitment to use a specific amount of compute power (measured in $/hour) for a one- or three-year period.

**On-Demand Capacity Reservations**

On-Demand Capacity Reservations enable you to reserve capacity for your Amazon EC2 instances in a specific AZ for any duration. This gives you the ability to create and manage capacity reservations independently from the billing discounts offered by Savings Plans or Regional Reserved Instances. By creating Capacity Reservations, you ensure that you always have access to EC2 capacity when you need it, for as long as you need it. You can create Capacity Reservations at any time, without entering into a one-year or three-year term commitment, and the capacity is available immediately. When you no longer need the reservation, we recommend that you cancel the Capacity Reservation to stop incurring charges for it.

**Instance Family Availability across AZs**

Certain Amazon EC2 instance families (for example X1 and High Memory) are not available across all AZs within a Region. You should confirm the instance types you require for your SAP workload are available in your target AZs when planning which ones, you intend to use.

**EC2 Auto-Recovery**

Amazon EC2 Auto Recovery is an Amazon EC2 feature that automatically recovers the instance within the same AZ if it becomes impaired due to an underlying hardware failure or a problem that requires AWS involvement to repair.

You can enable Auto Recovery for EC2 instances by creating an AWS CloudWatch alarm which monitors the instance status. Examples of problems that cause system status checks to fail include:

- Loss of network connectivity
- Loss of system power
- Software issues on the physical host
- Hardware issues on the physical host that impact network reachability

However, it typically takes under 15 minutes for a failed instance to restart, EC2 Auto Recovery does not offer a Service Level Goal. Therefore, if recovery of the application running (i.e. SAP Database or SAP
Central Services) on the failed host is critical you should consider using other approaches for providing High Availability such as clustering across two AZs.

**High Memory Dedicated Hosts**

Amazon EC2 High Memory Instances are specifically designed to run large in-memory databases such as SAP HANA. High Memory instances are available on EC2 Dedicated Hosts on a one- or three-year reservation.

High Memory instances support Dedicated Host Recovery. Host recovery automatically restarts your instances on to a new replacement host if failures are detected on your Dedicated Host. Host recovery reduces the need for manual intervention and lowers the operational burden if there is an unexpected Dedicated Host failure.

However, to protect against AZ failure we recommend you have a second-High Memory instance in a different AZ within your chosen Region. If you have a requirement for a non-production High Memory instance which is the same size as Production you can have this allocated to a different AWS account and use Dedicated Host Sharing to make it available within the production AWS account. In the event of an AZ failure impacting Production you can then take the non-production host and reallocate for production as a cost optimization measure.

**EC2 Maintenance**

When AWS maintains the underlying host for an instance, it schedules the instance for maintenance. There are two types of maintenance events:

- During network maintenance, scheduled instances lose network connectivity for a brief period of time. Normal network connectivity to your instance is restored after maintenance is complete.
- During power maintenance, scheduled instances are taken offline for a brief period, and then rebooted. When a reboot is performed, all of your instance's configuration settings are retained.

Additionally, we frequently upgrade our Amazon Elastic Compute Cloud (Amazon EC2) fleet, with many patches and upgrades being applied to instances transparently. However, some updates require a short reboot in order to apply these updates. Reboots such as these should be infrequent but may be necessary from time to time to apply upgrades that strengthen our security, reliability, and operational performance.

There are two kinds of reboots that can be required as part of Amazon EC2 scheduled maintenance:

- Instance reboots are reboots of your virtual instance and are equivalent to an operating system reboot.
- System reboots require reboots of the underlying physical server hosting an instance.

You can view any upcoming scheduled events for your instances in the AWS Management Console or using the API tools or command line.

If you do not take any action, the impact on your instance is the same in both cases – during your scheduled maintenance window your instance will experience a reboot that in most cases takes a few minutes.

Alternatively, you can stop and start the instance, which migrates it to a new host. For more information about stopping your instance, in addition to information about the changes to your instance configuration when it's stopped, see Stop and start your instance. You can automate an immediate stop and start in response to a scheduled maintenance event.
Networking

Amazon Virtual Private Cloud and Subnets

An Amazon Virtual Private Cloud (Amazon VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch your AWS resources, such as Amazon EC2 instances, into your VPC.

When you create a VPC, you must specify a range of IPv4 addresses for the VPC in the form of a Classless Inter-Domain Routing (CIDR) block; for example, 10.0.0.0/16. This is the primary CIDR block for your VPC.

You can create a VPC within your chosen AWS Region and it will be available across all AZs within that Region.

To add a new subnet to your VPC, you must specify an IPv4 CIDR block for the subnet from the range of your VPC. You can specify the AZ in which you want the subnet to reside. You can have multiple subnets in the same AZ but a single subnet cannot span across multiple AZs.

To provide future flexibility we recommend regardless of the number of AZs within a Region you initially plan to use that you ensure your subnet and connectivity design supports all the available AZs in your account within the Region.

Cross AZ latency

All AZs within a Region are connected through low-latency links and the typical latency between two AZs is 1 - 2 milliseconds.

For High Availability we recommend you deploy your Production SAP workloads across multiple AZs, including the SAP Application Server Layer. If you have SAP transactions or batch jobs that involve significant database calls then we recommend you run these transactions on SAP Application Servers located in the same AZ as the Database and use SAP Logon Groups (SMLG) for end users and batch server group (SM61) for Background Processing Jobs. This will ensure the latency sensitive parts of the SAP workload run on the correct application servers.

On-premises to AWS Connectivity

You can connect to your VPC through a Site to Site virtual private network (VPN) or AWS Direct Connect (DX) from on premises. Direct Connect provides and SLA of up to 99.99% and Site to Site VPN provides an SLA of 99.95%

Site to Site VPN connections are to specific Regions; for Direct Connect based connections, Direct Connect Gateway allows you to connect to multiple Regions.

When establishing connectivity to AWS from on-premises you should ensure you have resilient connections either through the use of multiple Direct Connect Links, Multiple VPN connections, or a combination of the two.

The AWS Direct Connect Resiliency Toolkit provides a connection wizard with multiple resiliency models. These models help you to order dedicated connections to achieve your SLA objective.

VPC Endpoints

A VPC endpoint enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by AWS PrivateLink without requiring internet access via an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC do not require public
IP addresses to communicate with resources in the AWS service. Traffic between your VPC and the other service does not leave the Amazon network.

VPC endpoints are available for all the core AWS services required to support an SAP based workload including Amazon EC2 API, Amazon S3, and Amazon Elastic File System.

**Cross-Region Peering**

Amazon Virtual Private Cloud (Amazon VPC) supports Inter-Region peering between two VPCs in different Regions. This can be used to allow network traffic such as database replication traffic to flow between two Amazon EC2 instances in different Regions. Use of Inter-Region peering will incur data transfer costs.

Amazon Transit Gateway (TGW) is a network transit hub that you can use to interconnect your virtual private clouds (VPC) within an AWS Region to other VPCs in other AWS Regions and on-premises networks using AWS Direct Connect or VPN. Use of Transit Gateway will incur Transit Gateway costs. Amazon Transit Gateway provides an SLA of 99.95% within a Region.

**Load balancing**


A Network Load Balancer can be used to support a High Availability deployment of SAP Web Dispatchers and/or SAP Central Services (see this document for more details) across multiple AZs.

A load balancer serves as the single point of contact for clients. The load balancer distributes incoming traffic across multiple targets, such as Amazon EC2 instances.

A listener checks for connection requests from clients, using the protocol and port that you configure, and forwards requests to a target group.

Each target group routes requests to one or more registered targets, such as EC2 instances, using the TCP protocol and the port number that you specify. You can configure health checks on a per target group basis. Health checks are performed on all targets registered to a target group that is specified in a listener rule for your load balancer.

For TCP traffic, the network load balancer selects a target using a flow hash algorithm based on the protocol, source IP address, source port, destination IP address, destination port, and TCP sequence number. Each individual TCP connection is routed to a single target for the life of the connection.

**DNS**

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. You can use Route 53 to perform three main functions in any combination: domain registration, DNS routing, and health checking. Amazon Route 53 offers an SLA of 100%.

Amazon Route 53 Resolver provides a set of features that enable bi-directional querying between on-premises and AWS over private connections.

**Storage**

**Object Storage**

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. Amazon S3 is a Regional service across all AZs within a Region and is designed for 99.999999999% (11 9's) of durability and an SLA of 99.9%. 
To protect against data loss, you can perform backups (such as database backups or file backups) to Amazon S3. Additionally Amazon EBS Snapshots and Amazon Machine Images (AMI) are stored in S3.

Amazon S3 Replication enables automatic, asynchronous copying of objects across Amazon S3 buckets. Buckets that are configured for object replication can be owned by the same AWS account or by different accounts.

**S3 replication**

You can replicate objects between different AWS Regions or within the same AWS Region.

- Cross-Region replication (CRR) is used to copy objects across Amazon S3 buckets in different AWS Regions.
- Same-Region replication (SRR) is used to copy objects across Amazon S3 buckets in the same AWS Region.

Cross-Region Replication will incur the following costs:

- Data Transfer charges for the data transferred between the First and Second AWS Regions
- Amazon S3 charges for the data stored in Amazon S3 in the two different AWS Regions

Additionally, with Cross-Region replication you can enable S3 Replication Time Control. S3 Replication Time Control (S3 RTC) helps you meet compliance or business requirements for data replication and provides visibility into Amazon S3 replication times. S3 RTC replicates most objects that you upload to Amazon S3 in seconds, and 99.99 percent of those objects within 15 minutes.

S3 Replication Time Control will incur the following costs in addition to the costs listed above for Cross-Region replication:

- S3 Replication Time Control Management Feature - priced per GB
- Amazon CloudWatch S3 Metrics - priced by number of metrics

Same-Region replication will incur the following costs:

- S3 Charges for the data stored in Amazon S3

**Block Storage**

Amazon Elastic Block Store (Amazon EBS) provides block level storage volumes for use with EC2 instances. EBS volumes behave like raw, unformatted block devices. You can mount these volumes as devices on your instances. You can create a file system on top of these volumes or use them in any way you would use a block device (like a hard drive). You can dynamically change the configuration of a volume attached to an instance.

Amazon EBS volumes are placed in a specific AZ where they are automatically replicated to protect you from the failure of a single component. All Amazon EBS volume types offer durable snapshot capabilities and are designed for 99.999% availability per volume and 99.99% service availability with Multi-AZ configuration. The use of a database replication capability, block level replication solution or EBS Snapshots is required to provide durability of the SAP data stored on EBS across multiple AZs.

Amazon EBS volumes are designed for an annual failure rate (AFR) of between 0.1% - 0.2%, where failure refers to a complete or partial loss of the volume, depending on the size and performance of the volume. This makes EBS volumes 20 times more reliable than typical commodity disk drives, which fail with an AFR of around 4%. For example, if you have 1,000 EBS volumes running for 1 year, you should expect 1 to 2 will have a failure.
Amazon EBS offers a number of different volume types, and for the SAP database-related data, General Purpose SSD (gp2) or Provisioned IOPS SSD (io1) must be used. The throughput and IOPS requirement will determine if gp2 or io1 is required.

Amazon EBS Multi-Attach enables you to attach a single Provisioned IOPS SSD (io1) volume to up to 16 AWS Nitro based instances that are in the same AZ. You can attach multiple Multi-Attach enabled volumes to an instance or set of instances. Each instance to which the volume is attached has full read and write permission to the shared volume. Multi-Attach enabled volumes do not support I/O fencing. I/O fencing protocols control write access in a shared storage environment to maintain data consistency. Your applications must provide write ordering for the attached instances to maintain data consistency.

EBS Snapshots

You can back up the data on your Amazon EBS volumes to Amazon S3 by taking point-in-time snapshots. Snapshots are incremental backups, which means that only the blocks on the device that have changed after your most recent snapshot are saved. This minimizes the time required to create the snapshot and saves on storage costs by not duplicating data. When you delete a snapshot, only the data unique to that snapshot is removed. Each snapshot contains all of the information that is needed to restore your data (from the moment when the snapshot was taken) to a new EBS volume.

EBS Snapshots can be copied (replicated) to a different Region and / or shared with a different AWS Account.

Copying Snapshots across Regions will incur the following costs:-

- Data Transfer Charges for the data transferred between the First and Second AWS Regions
- EBS Snapshot Charges for the data stored in Amazon S3 in the two different AWS Regions

Restoring Snapshots

New volumes created from existing EBS snapshots load lazily in the background. This means that after a volume is created from a snapshot, there is no need to wait for all of the data to transfer from Amazon S3 to your EBS volume before your attached instance can start accessing the volume and all its data.

This preliminary action takes time and can significantly increase the latency of I/O operations. If your instance accesses data that hasn't yet been loaded, the volume immediately downloads the requested data from Amazon S3, and then continues loading the rest of the volume data in the background.

Fast Snapshot Restore

Amazon EBS fast snapshot restore enables you to create a volume from a snapshot that is fully-initialized at creation. This eliminates the latency of I/O operations on a block when it is accessed for the first time. Volumes created using fast snapshot restore instantly deliver all of their provisioned performance. To use fast snapshot restore, enable it for specific snapshots in specific AZs. Fast Snapshot Restore is charged in Data Services Unit-Hours (DSUs) for each AZ in which it is enabled. DSUs are billed per minute with a 1 hour minimum.

File Storage

Amazon EFS

Amazon Elastic File System (EFS) provides scalable NFS version 4 based file storage for use with Linux based Amazon EC2 (Windows based EC2 instances do not support EFS). The service is designed to be highly scalable, highly available, and highly durable. Amazon EFS file systems store data and metadata across multiple AZs in an AWS Region. Amazon EFS offers an SLA of 99.99%.

EFS file systems can be shared across accounts and VPCs within the same Region or a different Region, enabling EFS to be an ideal choice for SAP global file system (/sapmnt) and SAP transport directory (/usr/sap/trans).
AWS DataSync supports EFS to EFS transfer between Regions and different AWS Accounts to allow the replication of key SAP file based data across Regions. AWS Backup can also be used to replicate backups of EFS file systems across Regions.

Amazon FSx

Amazon FSx for Windows File Server provides fully managed Microsoft Windows file servers, backed by a fully native Windows file system. Amazon FSx offers an SLA of 99.9% and supports both Single-AZ and Multi-AZ File Systems.

With Single-AZ file systems, Amazon FSx automatically replicates your data within an AZ to protect it from component failure, continuously monitors for hardware failures, and automatically replaces infrastructure components in the event of a failure. Amazon FSx also takes highly durable backups of your file system daily using Windows' Volume Shadow Copy Service that are stored in Amazon S3, and it allows you to take additional backups at any point.

Multi-AZ file systems support all the availability and durability features of Single-AZ file systems, and in addition, are designed to provide continuous availability to data, even in the event that an AZ is unavailable. In a Multi-AZ deployment, Amazon FSx automatically provisions and maintains a standby file server in a different AZ. Any changes written to disk in your file system are synchronously replicated across AZs to the standby.

FSx File systems can be shared across Accounts and VPCs within the same Region or a different Region, enabling FSx to be used not only for the SAP Global File System but also the SAP Transport Directory. Additionally, Amazon FSx can also be used for providing Continuously Available (CA) File Shares for Microsoft SQL Server.

Monitoring and Audit

Amazon CloudWatch

Amazon CloudWatch is a monitoring and observability service built for DevOps engineers, developers, site reliability engineers (SREs), and IT managers. CloudWatch provides you with data and actionable insights to monitor your applications, respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health. CloudWatch collects monitoring and operational data in the form of logs, metrics, and events, providing you with a unified view of AWS resources, applications, and services that run on AWS and on-premises servers. You can use CloudWatch to detect anomalous behavior in your environments, set alarms, visualize logs and metrics side by side, take automated actions, troubleshoot issues, and discover insights to keep your applications running smoothly.

AWS CloudTrail

AWS CloudTrail is a service that enables governance, compliance, operational auditing, and risk auditing of your AWS account. With CloudTrail, you can log, continuously monitor, and retain account activity related to actions across your AWS infrastructure. CloudTrail provides event history of your AWS account activity, including actions taken through the AWS Management Console, AWS SDKs, command line tools, and other AWS services. This event history simplifies security analysis, resource change tracking, and troubleshooting. In addition, you can use CloudTrail to detect unusual activity in your AWS accounts. These capabilities help simplify operational analysis and troubleshooting.

Architecture Patterns

In this section we detail the architecture options dependent on your availability and recovery requirements as well as the failure scenarios to consider when selecting the right architecture for your SAP system(s).
Failure Scenarios

Before looking at the architecture options, first consider the potential failure scenarios and the impact on the availability of an SAP system.

For the failure scenarios below the primary consideration is the physical unavailability of the compute and / or storage capacity within the AZs.

Availability Zone Failure

An AZ failure would be the result of a significant degradation of the availability of one or more AWS services that your resources depend on within that AZ.

Examples include:

- A significant number of Amazon Elastic Compute Cloud (Amazon EC2) instances have failed with System Status Check errors or are unreachable and cannot be restarted within the AZ
- A significant number of Amazon Elastic Block Store (EBS) volumes with Volume Status Check errors

Amazon Elastic Block Store failure

Loss of one or more EBS volumes attached to a single EC2 instance may result in the unavailability of a critical component (i.e. the database) of the SAP system.

EC2 failure

Loss of a single EC2 instance may result in the unavailability of a critical component (i.e. the database or SAP central services) of the SAP system.

Logical Data Loss

You should also consider the potential for logical data loss where the underlying hardware capacity still exists but the primary copies of the data have been corrupted or lost. This data loss could be due to malicious activity within your AWS account or due to human error.

To protect against logical data loss, it is recommended that regular copies of the data are backed up to an S3 bucket and this bucket is replicated (using Single-Region or Cross-Region replication (p. 91) depending on your requirements) to another Amazon S3 bucket owned by a separate AWS account. This approach ensures that with the appropriate AWS Identity and Access Management (IAM) controls between the two AWS accounts it is not possible for all copies of the data to be lost through malicious activity or due to human error.

Patterns

In this section we examine the architecture patterns available to handle the failure scenarios detailed above.

When evaluating each pattern and its fit for your organization’s specific business requirements, the two key resources to consider are availability of compute for the SAP single points of failure (Database and SAP Central Services) and the availability of the SAP data persisted on block storage (EBS), as these will determine the time taken to recover from the failure scenario - in other words, the time to return the SAP system to service.

We have grouped the patterns into single Region and multi-Region patterns. When evaluating each pattern, the following will be key to determine which pattern(s) are suitable for your organization:
1. You require the data to only reside in a specific geographical location (AWS Region) at all times (for example due to data residency requirements) or

2. You require the data to reside in two specific geographical locations (AWS Regions) at all times (for example two copies of the SAP data must reside at least 500 miles apart for compliance reasons)

If you have a requirement for an uptime of >= 95% and a short time to return the SAP system to service then you will need to consider a multi-AZ pattern. However, if you do not have a requirement for the data to reside in two specific geographical locations (AWS Regions) then you can leverage the multi-AZ pattern for High Availability to also provide Disaster Recovery and therefore lower the overall costs.

**Single Region Architecture Patterns**

You would select a single Region architecture based on the following decision points:

- You require the data to only reside in a specific geographical Region (AWS Region) at all times
- You wish to avoid the potential network latency (p. 85) considerations associated with a multi-Region approach
- You wish to avoid the cost implications or differences associated with a multi-Region approach including:
  - AWS service pricing in different AWS Regions (p. 85)
  - Cross-Region data transfer costs (p. 85)

**Pattern 1: A Single Region with a Single AZ for Production**

![Figure 7: Single Region with a single AZ for Production](image)

In this pattern, you deploy all your production systems in one AZ and all your non-production systems in either the same AZ or another AZ. Your non-production systems are not a similar size to your production.

**When to consider Pattern 1**

- In the event of an AZ failure or significant Amazon EC2 service degradation, you can accept the variable time duration required (including any delay in availability of the required compute capacity in the remaining AZs) to re-create the AWS resources in a different AZ and restore the persistent data to EBS.
- You wish to avoid the cost implications with a multi-AZ approach related to Cross-AZ network traffic costs.
Key design principles

- 100% compute capacity deployed in AZ 1 for Production SAP database and central services tiers
- 100% compute capacity deployed in AZ 1 for Production SAP Application Tier
- EC2 auto recovery (p. 87) is configured for all instances to protect against underlying hardware failure
- Deployed non-Production compute capacity is less than 100% the compute capacity deployed for Production SAP database and central services tiers
- EFS used for the SAP Global File Systems
- The SAP Database data is persisted on EBS in a single AZ only and not replicated to another AZ using either a database replication capability or a block level replication solution
- SAP Database data is backed up regularly to S3
- Amazon S3 single-Region replication (p. 91) is configured to protect logical data loss (p. 94)
- Amazon Machine Image / EBS Snapshots for all servers are taken on a regular basis

Benefits

- Lowest Cost
- Simplest Design
- Simplest Operation

Considerations

- Well documented and tested processes for re-creating the required AWS resources and restoring the Data in a different AZ are required to ensure recoverability

Pattern 2: A Single Region with one AZ for Production and another AZ for Non-Production

In this pattern, you deploy all your production systems in one AZ and all your non-production systems in another AZ. Your non-production systems are an equivalent size to your production.

When to consider Pattern 2

- You require a defined time window to complete recovery of Production and assurance of the availability of the compute capacity in another AZ for the SAP database and central services tiers
You can accept the additional time required to re-allocate compute capacity from non-production to production as part of the overall time window to recover Production.

You can accept the time required to restore data to EBS from Amazon S3 in another AZ as part of the overall time window to recover Production.

You can accept the variable time duration required to return the application tier to 100% capacity following an AZ failure (including any delay in availability of the required compute capacity in the remaining AZs).

You can accept a period of time where there is only one set of computes deployed for the Production SAP database and central services tiers in the event of an AZ failure or significant EC2 service degradation.

**Key design principles**

- 100% compute capacity deployed in AZ 1 for Production SAP database and central services tiers.
- 100% compute capacity deployed in AZ 1 for Production SAP Application Tier.
- 100% of Production compute capacity (SAP database and central services) deployed in AZ 2 for use by Non-Production in normal operations.
- EC2 auto recovery (p. 87) is configured for all instances to protect against underlying hardware failure.
- EFS used for the SAP Global File Systems.
- The SAP Database data is persisted on EBS in a single AZ only and not replicated to another AZ using either a database replication capability or a block level replication solution.
- SAP Database Data is backed up regularly to S3.
- Amazon S3 single-Region replication (p. 91) is configured to protect logical data loss (p. 94).
- Amazon Machine Image / EBS Snapshots for all servers are taken on a regular basis.

**Benefits**

- Cost optimized through use of non-production capacity in the event of Production AZ failure.
- Required Compute capacity deployed in two AZs to allow a more predictable recovery time duration.

**Considerations**

- Well documented and tested processes for re-allocating the required compute capacity from non-production to production and restoring the data in a different AZ are required to ensure recoverability.
- Loss of the non-production environments in the event of an AZ failure impacting production.
Pattern 3: A Single Region with two AZs for Production

In this pattern, which builds upon pattern 2, you deploy all your production systems across two AZs. The compute deployed for the production SAP database and central services tiers are the same size in both AZs with automated failover in the event of AZ failure. The compute required for the SAP application tier is split 50/50 between two AZs. Your non-production systems are not an equivalent size to your production and are deployed in the same AZs or a different AZ within the Region.

When to consider Pattern 3

You require the capabilities included in Pattern 2, including the following:

- You require a defined time window to complete recovery of Production and assurance of the availability of the compute capacity in another AZ for the Production SAP database and central services tiers
- You can accept the additional cost of deploying the required compute and storage for Production SAP database and central services tiers across two AZs
- Your non-Production environment is not equivalent size as Production and therefore cannot be used as sacrificial capacity for Production in the event of an AZ failure or significant EC2 service degradation
- You can accept that data replication across AZs of the Database related data requires either a database replication capability or a block level replication solution
- You can accept Cross-AZ related data transfer costs for Data replication
- You can accept that automated failover between AZs will requires 3rd party ISV product

Key design principles

The key design principles of Pattern 2, including the following:

- 100% compute capacity deployed in AZ 1 and AZ 2 for Production SAP database and central services tiers
- Compute capacity deployed in AZ 1 and AZ 2 for Production application tier (Active/Active). In the event of an AZ failure the application tier would need to be scaled to return to 100% capacity within the reminding AZ
- The SAP Database data is persisted on EBS in two AZs using either a database replication capability or a block level replication solution

Benefits

All the benefits included in Pattern 2, including the following:
- Database related data persisted on different sets of EBS volumes in two AZs
- Required compute capacity deployed in two AZs
- No requirement on restoring data from S3 in the event of an AZ failure
- Ability to protect against significant degradation or total AZ failure through failover of database and central services tiers to AZ 2

**Considerations**

- Well documented and tested processes for re-creating the required AWS resources to return the application tier to required capacity in the event of an AZ failure or significant EC2 service degradation

**Pattern 4: A Single Region with two AZs for Production and Production Sized Non-Production in a 3rd AZ**

In this pattern, which builds upon pattern 3, you deploy all your production systems across two AZs. The compute deployed for the production SAP database and central services tiers are the same size in both AZs with automated failover in the event of AZ failure. The compute required for the SAP application tier is split 50/50 between two AZs.

Your non-production systems are an equivalent size to your production and deployed in the third AZ. In the event of an AZ failure where your production systems are deployed, the non-production capacity is reallocated to enable Production to be returned to a multi-AZ pattern.

**When to consider this Pattern**

You require the capabilities provided by Pattern 3, including the following:

- You require the ability to continue to have a multi-AZ configuration for Production in event of an AZ failure within the Region

**Key design principles**

Includes the key design principles of Pattern 3, including the following:

- 100% of Production compute capacity (database and central services) deployed in three AZs for use by non-Production in normal operations

**Benefits**
All the included benefits of Pattern 3, including the following:

- Option for data to be persisted on EBS in three different AZs dependent on capabilities of database or block level replication solution
- Required compute capacity deployed in three AZs
- No requirement on restoring data from S3 in the event of an AZ failure
- Use of non-production compute capacity to return to Production run across two AZs in the event of a significant degradation or total AZ failure

Considerations

- Well documented and tested processes for re-creating the required AWS resources to return the application tier to required capacity in the event of an AZ failure or significant EC2 service degradation
- Well documented and tested processes for re-allocating the required compute capacity from non-production to return production to run across two AZs in the event of an AZ failure impacting production

Multi-Region Architecture Patterns

You should select a multi-Region architecture if you require the following:

- You require the data to reside in two specific geographical AWS Regions at all times
- You can accept the potential network latency considerations associated with a multi-Region approach
- You can accept the increased complexity associated with multi-Region approach
- You can accept the cost implications / differences associated with a multi-Region approach including:
  
  - **AWS service pricing (e.g. EC2) (p. 85)** in different AWS Regions
  - **Cross-Region data transfer costs (p. 85)**
  - **Additional compute and / or storage costs in the second Region**
Pattern 5: A Primary Region with one AZ for Production and Secondary Region containing a replica of backups / AMIs

Figure 11: A Primary Region with one AZ for Production and Secondary Region containing a replica of backups/AMIs

In this pattern, you deploy all your production systems in the primary Region in one AZ.

Your non-production systems are not an equivalent size to your production and are deployed in the same AZs or a different AZ within the Region.

Additionally, the production database backups stored in Amazon S3, EBS Snapshots, and Amazon Machine Images are replicated to a secondary Region. In the event of a complete Region failure, the production systems would be restored from the last set of backups in the second Region.

When to consider this Pattern

- In the event of an AZ failure or significant Amazon EC2 service degradation, you can accept the variable time duration required (including any delay in availability of the required compute capacity in the remaining AZs) to re-create the AWS resources in a different AZ and restore the persistent data to EBS.
- You can accept the variable time duration required to complete recovery of Production in the event of a Region failure
- You can accept the increased complexity and costs associated with multi-Region approach
- You can accept that manual actions are required to restore production in the secondary Region

Key design principles

- 100% compute capacity deployed in AZ 1 for Production SAP database and central services tiers
• 100% compute capacity deployed in AZ 1 for Production SAP Application Tier
• EC2 Auto-Recovery (p. 87) is configured for all instances to protect against underlying hardware failure
• Deployed non-Production compute capacity is less than 100% the compute capacity deployed for Production SAP database and central services tiers
• Amazon EFS used for the SAP global file systems
• The SAP Database data is persisted on EBS in a single AZ only and not replicated to another AZ using either a database replication capability or a block level replication solution
• SAP Database data is backed up regularly to Amazon S3
• Amazon S3 cross-Region replication (p. 91) is configured to protect logical data loss (p. 94)
• Amazon Machine Image / EBS Snapshots for all servers are taken on a regular basis
• Amazon S3 Data (database backups), EBS Snapshots and Amazon Machine Images are replicated to a secondary Region to protect logical data loss (p. 94)

Benefits

• Reduced cost
• Ability to protect against significant degradation or total Region failure through failover to secondary Region

Considerations

• Well documented and tested processes for re-creating the required AWS resources to return the SAP application tier to required capacity in the event of an AZ failure or significant EC2 service degradation
• Well documented and tested processes for re-creating the required AWS resources, restore the data and move Production to the secondary Region
• Network latency from your on-premises locations to the secondary AWS Region
Pattern 6: A Primary Region with two AZs for Production and Secondary Region containing a replica of backups / AMIs

In this pattern, you deploy all your production system across two AZs in the primary Region. The compute deployed for the production SAP database and central services tiers are the same size in both AZs with automated failover in the event of AZ failure. The compute required for the SAP application tier is split 50/50 between two AZs.

Your non-production systems are not an equivalent size to your production and are deployed in the same AZs or a different AZ within the Region.

In addition, the Production database backups stored in Amazon S3, EBS Snapshots, and Amazon Machine Images are replicated to the secondary Region. In the event of a complete Region failure the production systems would be restored from the last set of backups in the second Region.

When to consider this Pattern

- You require a defined time window to complete recovery of Production in the event of an AZ failure or significant Amazon EC2 failure
- You require assurance of the availability of the compute capacity in another AZ in the primary Region for the Production SAP database and central services tiers
- You can accept the variable time duration required to complete recovery of Production in the event of a Region failure
- You can allow for a period of time where there is only one set of computes deployed for the SAP database and central services in the event of a AZ failure or significant EC2 failure
- You can accept the increased cost of deploying the required compute and storage for Production SAP database and central services tiers across two AZs
• You can accept that replication across AZs of the Database requires either a database replication capability or a block level replication solution
• You can accept the increased complexity and costs associated with multi-Region approach
• You can accept the cross-AZ related data transfer costs for Data replication
• You can accept that automated failover between AZs will requires 3rd party ISV clustering solution
• You can accept the variable time duration required (including any delay in availability of the required compute capacity in the remaining AZs) return application tier to 100% capacity
• You can accept that manual actions are required to restore production in the second Region

Key design principles
• 100% Compute Capacity Deployed in AZ 1 and AZ 2 for Production SAP database and central services tiers
• Compute Capacity deployed in AZ 1 and AZ 2 for Production SAP application tier (Active/Active) and would need to be scaled in the event of an AZ failure or significant EC2 service degradation
• EC2 Auto-Recovery (p. 87) is configured for all instances to protect against underlying hardware failure with the exception of those instances protected with a 3rd party ISV clustering solution
• The SAP Database data related data on EBS replicated between AZs using either a database replication capability or a block level replication solution
• EFS used for the SAP Global File Systems
• SAP Database Data is backed up regularly to Amazon S3
• Amazon Machine Image / EBS Snapshots for all servers are taken on a regular basis
• Amazon S3 Data (database backups), EBS Snapshots, and Amazon Machine Images are replicated / copied to a secondary Region to protect logical data loss (p. 94)

Benefits
• Low Mean Time to Recovery (MTTR)
• Predictable Return to Service (RTS)
• Database related data persisted on different sets of EBS volumes in two AZs via database replication capability or a block level replication solution
• Required compute capacity deployed in two AZs
• No dependency on restoring data from S3 in the event of an AZ failure in the primary Region
• Ability to protect against significant degradation or total AZ failure through failover to AZ2 of database and central services tiers
• Ability to protect against significant degradation or total Region failure through failover to secondary Region

Considerations
• Well documented and tested processes for re-creating the required AWS resources to return the SAP application tier to required capacity in the event of an AZ failure or significant EC2 service degradation
• Well documented and tested processes for re-creating the required AWS resources, restore the data and move Production to the secondary Region
• Network latency from your on-premises locations to the secondary AWS Region
Pattern 7: A Primary Region with two AZs for Production and Secondary Region with Compute and Storage capacity deployed in a Single AZ

In this pattern, which builds upon pattern 6, you deploy all of your production systems across two AZs in the primary Region. The compute deployed for the production SAP database and central services tiers are the same size in both AZs with automated failover in the event of AZ failure. The compute required for the SAP application tier is split 50/50 between two AZs. Your non-production systems are not an equivalent size to your production and are deployed in a different AZ within the Region.

In addition, you have compute capacity deployed in AZ 1 in Secondary Region for production SAP database and central services tiers and the production database is replicated to the secondary Region using either a database replication capability or a block level replication solution.

The Production database backups stored in S3, EBS Snapshots, and Amazon Machine Images in the primary Region are replicated to the secondary Region. In the event of a complete Region failure the production systems would be restored using the replicated data for the Database tier and the last set of backups for the SAP Central Services and Application tiers.

When to consider this Pattern

You require the capabilities included in Pattern 6, including the following:

- You require a defined time window to complete recovery of Production in the event of a Region failure
- You require assurance of availability of compute capacity in a single AZ in the secondary Region for the Production SAP database and central services tiers
• You can accept the increased cost of deploying the required compute and storage for Production SAP database and central services tiers across two AZs in the primary Region and one AZ in the secondary Region

• You can accept that manual actions are required to failover between Regions

**Key design principles**

You require the key design principles included in Pattern 6, including the following:

• 100% compute capacity deployed in AZ 1 in the secondary Region for Production SAP database and central services tiers

• The SAP Database data related data on EBS replicated between Regions using either a database replication capability or a block level replication solution

• EFS used for the SAP Global File Systems and replicated to the secondary Region

**Benefits**

You require all the benefits included in Pattern 6, including the following:

• Low Mean Time to Recovery (MTTR) in the event Region failure

• Database related data persisted on different sets of EBS volumes in two AZs in primary Region and one set of volumes in an AZ in secondary Region via database replication capability or a block level replication solution

• Required compute capacity deployed in two AZs in primary Region and one AZ in secondary Region

• No dependency on restoring data from Amazon S3 in the event of an AZ failure or Region failure

• Ability to protect against significant degradation or total AZ failure through failover to AZ2 of database and central services tiers

• Ability to protect against significant degradation or total Region failure through failover to secondary Region

**Considerations**

• Well documented and tested processes for re-creating the required AWS resources to return the application tier to required capacity in the event of an AZ failure or significant EC2 service degradation

• Well documented and tested processes for moving production to the secondary Region
Pattern 8: A Primary Region with two AZs for Production and Secondary Region with Compute and Storage capacity deployed and Data Replication across two AZs

In this pattern, which builds upon pattern 7, you deploy all of your production systems across two AZs in the primary Region. The compute deployed for the production SAP database and central services tiers are the same size in both AZs with automated failover in the event of AZ failure. The compute required for the SAP application tier is split 50/50 between two AZs. Your non-production systems are not an equivalent size to your production and are deployed in the same AZs or a different AZ within the Region.

In addition, the production database backups stored in Amazon S3, EBS Snapshots, and Amazon Machine Images are replicated to a secondary Region and you have compute capacity deployed in AZ 1 and AZ 2 in secondary Region for production SAP database and central services tiers.

When to consider this Pattern

You require the capabilities included in Pattern 7, including the following:

- You require assurance of availability of compute capacity in two AZs in the secondary Region for the Production SAP database and central services tiers
- You can accept the increased cost of deploying the required compute and storage for Production SAP database and central services tiers across two AZs in the primary Region and two AZs in the secondary Region
- You can accept that manual actions are required to failover between Regions
- You can accept the variable time duration required (including any delay in availability of the required compute capacity in the remaining AZs) to return the application tier to 100% capacity
Key design principles

You require the key design principles included in Pattern 7, including the following:

• 100% compute capacity deployed in AZ 1 and AZ 2 in secondary Region for Production SAP database and central services tiers

Benefits

You require all the benefits included in Pattern 7, including the following:

• Database related data persisted on different sets of EBS volumes in two AZs in the primary Region and different sets of EBS volumes in two AZs in the secondary Region via database replication capability or a block level replication solution
• Required compute capacity deployed in two AZs in primary Region and two AZs in secondary Region
• Ability to protect against significant degradation or total AZ failure through failover to AZ 2 of database and central services tiers in both Regions
• Ability to protect against significant degradation or total Region failure through failover to secondary Region

Considerations

• Well documented and tested processes for re-creating the required AWS resources to return the application tier to required capacity in the event of an AZ failure or significant EC2 service degradation
• Well documented and tested processes for moving production to the secondary Region
• You will need to maintain clustering solutions in both Regions and the associated costs of this

Summary

The table below summarizes the patterns and their key characteristics.

<table>
<thead>
<tr>
<th>#</th>
<th>Single Region</th>
<th>Multi Region</th>
<th>Single AZ Primary</th>
<th>Multi AZ Primary</th>
<th>Single AZ Second Region</th>
<th>Multi AZ Second Region</th>
<th>Prod Capacity in 2nd AZ</th>
<th>Use of non-prod capacity</th>
<th>Cross Region data replication</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>8</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 1: Summary of Patterns
With the flexibility and agility of the AWS cloud you have the ability to select one of patterns we have described above per SAP system you intend to deploy in AWS based on the specific requirements per system rather than taking the highest requirement and applying this to all production systems.

For example:

- For your core ERP system, you require the assurance of the availability of the compute capacity in another AZ for the Production SAP database and central services tiers.
- For your BW system, you can accept the variable time duration required to re-create the AWS resources in a different AZ and restore the persistent data.

In this example, you would select Pattern 3 for ERP and Pattern 1 for BW to reduce the overall TCO.

It is also important to remember that should your requirements change over time it is possible to move to a different pattern without significant re-design. For example, during the earlier phases of an implementation project you may not require the assurance of the availability of the compute capacity in another AZ for the Production SAP database and central services tiers and can deploy the capacity into the second AZ a few weeks before the time it is required for testing before go-live.

When selecting the architecture pattern(s) that most suits your requirements it is important to consider the following key factors for each SAP system you plan to run in AWS and select the most appropriate pattern that meets those requirements.

- The geographical residency of the data you require
- The recovery time objective you require
- The recovery point objective you require
- The cost profile you require

**Additional Reading**

- SAP on AWS Technical Documentation
- SAP on AWS Overview and Planning
- SAP on AWS Pricing and Optimization
- Capacity planning for SAP on AWS whitepaper
- Backup and recovery whitepaper
- AWS Documentation
  - Amazon Web Services: Overview of Security Processes
  - AWS Well-Architected Framework – Reliability Pillar

**Document Revisions**

<table>
<thead>
<tr>
<th>Date</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>June 2020</td>
<td>Initial publication</td>
</tr>
</tbody>
</table>
About This Guide

This guide provides instructions on how to set up Amazon Web Services (AWS) resources to deploy IBM Db2 High Availability Disaster Relief (HADR) with Pacemaker for SAP NetWeaver on Amazon Elastic Compute Cloud (Amazon EC2) instances. This guide is for users who are responsible for planning, architecting and deploying IBM Db2 on AWS for SAP NetWeaver-based applications.

Overview

Instructions in this document are based on recommendations provided by SAP and IBM on Db2 deployment on Linux via the SAP notes and KB articles listed in Table 1.

Table 1 - SAP NetWeaver on IBM Db2 OSS Notes

<table>
<thead>
<tr>
<th>SAP OSS Note</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1656099</td>
<td>SAP Applications on AWS: Supported DB/OS and AWS EC2 products</td>
</tr>
<tr>
<td>1656250</td>
<td>SAP on AWS: Supported instance types</td>
</tr>
<tr>
<td>1612105</td>
<td>DB6: FAQ on Db2 High Availability Disaster Recovery (HADR)</td>
</tr>
<tr>
<td>101809</td>
<td>DB6: Supported Db2 Versions and Fix Pack Levels</td>
</tr>
<tr>
<td>1168456</td>
<td>SAP Db2 support info</td>
</tr>
<tr>
<td>1600156</td>
<td>SAP Db2 support on AWS</td>
</tr>
</tbody>
</table>

This document follows best practices from AWS, IBM and SAP for SAP NetWeaver deployments on Linux. See the Additional Reading (p. 153) section of this document for more detail.

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the Amazon Web Services (AWS) Cloud. For the other guides in the series, ranging from overviews to advanced topics, see https://aws.amazon.com/sap/docs/

What this guide doesn't do

This document doesn't provide guidance on how to set up network and security constructs like Amazon Virtual Private Cloud (Amazon VPC), subnets, route tables, access control lists (ACLs), Network Address
Translation (NAT) Gateway, Identity and Access Management (IAM) Roles, or AWS Security Groups. It doesn’t cover the high availability (HA) setup for the SAP Application Server Central Services/Evaluated Receipt Settlement (ASCS/ERS), and focuses only on the database (DB) layer when covering the single points of failure (SPOF) for the SAP applications.

**Considerations**

**Specialized Knowledge**

To understand this document, you should have a good understanding of AWS services, general networking concepts, Linux operating systems, and IBM Db2 administration.

Before you follow the instructions in this guide, we recommend that you become familiar with the following AWS services. (If you are new to AWS, see [Getting Started with AWS](#).

- Amazon EC2
- Amazon EBS
- Amazon VPC
- AWS CloudFormation
- AWS Systems Manager
- Amazon Simple Storage Service (Amazon S3)
- AWS Identity and Access Management (IAM)

**Technical Requirements**

- Before you start the installation and configuration of IAM Db2 High Availability Disaster Recovery (HADR), ensure that you meet the following requirements:
- Your operating system is a supported Red Hat or SUSE version. Check SAP product availability matrix (PAM). Login required.
- Your database version is IBM Db2 10.5 or higher.
- Bring your own license (BYOL) for IBM Db2 and SAP application.
- Install AWS SAP Data Provider on Amazon EC2 instances after installing IBM Db2 database.
- An AWS account with permission to create resources.
- Access to SAP installation media for database and application.
- AWS Business or Enterprise level support (1656250 - SAP on AWS: Support prerequisites). Login required.

**Planning**

**Architecture Options**

SAP NetWeaver applications based on IBM Db2 can be installed in three different ways:

- **Standard system or single host installation**— In this option, Advanced Business Application Programming (ABAP) Application Server Central Services/System Central Services (ASCS/SCS) and the database primary application server (PAS) of SAP NetWeaver run in a single Amazon EC2 instance. This option is suited for non-critical and non-production workloads.
• **Distributed system**— In distributed systems, ASCS/SCS and the database PAS of SAP NetWeaver can run on separate Amazon EC2 instances. For example, you can choose to run ASCS and PAS on one Amazon EC2 instance, and the database on another Amazon EC2 instance, or other combinations. This option is suited for production and non-production workloads.

• **High availability system**— For your SAP application to be highly available, you will need to protect the single point of failures. The database is one of the single points of failure in SAP applications.

AWS recommends that you deploy primary and standby IBM Db2 databases in different Availability Zones (AZs) within an AWS region. Figure 1 provides a high-level architecture for IBM Db2 high availability in AWS. This option is suited for business-critical applications.

![Figure 1 – High-level HA architecture for SAP with IBM Db2 on AWS](image)

**Security**

AWS provides security capabilities and services to securely run your SAP applications on the AWS platform. In the context of IBM Db2 for SAP applications, you can use network services and features such as Amazon VPC, AWS Virtual Private Network (AWS VPN), AWS Direct Connect, Amazon EC2 Security Groups, network access control lists (NACLs), route tables, and more to restrict the access to your database.

**Network Security**

The databases of SAP applications don’t usually require direct user access. The end users access the application using SAP Graphical User Interface (GUI), SAP Web Dispatcher, or SAP Fiori. We recommend that you limit direct access to the EC2 instances to administrators only, for maintenance purpose.

IBM Db2 listens on TCP port 5912 by default. Depending on your VPC design, you should configure Amazon EC2 Security Groups, network and cryptography libraries (NACLs), and route tables to allow traffic to TCP Port 5912 from SAP primary application servers and additional application servers (PAS/AAS) and ABAP SAP Central Services/SAP Central Services (ASCS/SCS). To learn more about configuring the security group, see Security groups for your VPC.
Encryption

Encryption is a security mechanism that converts plain text (readable data) into ciphertext. AWS offers built-in encryption for Amazon EBS data volumes, boot volumes, and snapshots. The encryption process occurs automatically, and you don’t need to manage encryption keys. This mechanism protects your EBS volumes at rest, and data in transit that passes between EC2 servers. This encryption level is offered at no additional cost.

You also can use the native IBM Db2 native database encryption feature if required.

Sizing

SAP Quick Sizer is used to size SAP environment for new implementations. However, if you are migrating your existing SAP applications based on IBM Db2 to AWS, consider using the following tools to right-size your SAP environment based on current utilization.

- **SAP Early Watch Alerts (EWA):**—SAP EWA reports are provided by SAP regularly. These reports provide an overview of historical system utilization. Analyze these reports to see if your existing SAP system is over-utilized or under-utilized. Use this information to right-size your environment.

- **Linux native tools:**—Gather and analyze historical utilization data for CPU/Memory to right-size your environment. In case your source is IBM AIX, you can make use of nmon reports as well.

- **AWS Services**— Use services such as AWS Migration Evaluator or AWS Application Discovery Services that help with collecting usage and configuration data about your on-premises servers. Use this information to analyze and right-size your environment.

Because it’s easy to scale up or scale down your Amazon EC2 instances on AWS, consider the following while sizing your SAP environment on AWS.

- You don’t need to over-provision storage to meet future demand.

- SAP Quick Sizer tools provide sizing guidance based on assumptions that on 100% load (as per your inputs to tool), system utilization will not be more than 65%, so there is some buffer built into SAP Quick Sizer recommendation. See SAP’s [Quick Sizer guidance](#) for details. (Login required.)

Operating System

You can deploy your SAP workload on SUSE Linux Enterprise Server (SLES) for SAP, Red Hat Enterprise Linux for SAP with High Availability and Update Services (RHEL for SAP with HA and US), or RHEL for SAP Solutions.

SLES for SAP and RHEL for SAP with HA and US are available in the AWS Marketplace under an hourly or an annual subscription model.

**SLES**

SLES for SAP provides additional benefits, including Extended Service Pack Overlap Support (ESPOS), configuration and tuning packages for SAP applications, and High Availability Extensions (HAE). See the [SUSE SLES for SAP product page](#). We strongly recommend using SLES for SAP instead of SLES for all your SAP workloads.

If you plan to use Bring Your Own Subscription (BYOS) images provided by SUSE, ensure that you have the registration code required to register your instance with SUSE to access repositories for software updates.
RHEL

RHEL for SAP with HA and US provides access to Red Hat Pacemaker cluster software for High Availability, extended update support, and the libraries that are required to configure pacemaker HA. For details, see the RHEL for SAP Offerings on AWS FAQ in the Red Hat knowledgebase.

If you plan to use the BYOS model with RHEL, either through the Red Hat Cloud Access program or another means, ensure that you have access to a RHEL for SAP Solutions subscription. For details, see Overview of the Red Hat Enterprise Linux for SAP Solutions subscription in the Red Hat knowledgebase.

The correct subscription is required to download the required packages for configuring the Pacemaker cluster.

Compute

AWS provides a wide array of SAP supported Amazon EC2 instances for your SAP workloads. See SAP Note 1656099 - SAP Applications on AWS: Supported DB/OS and AWS EC2 products for details. Based on the results of your sizing exercise, you can deploy your IBM Db2 on any of the SAP supported Amazon EC2 instances that meets your requirement.

Storage

Amazon EBS volumes are designed to be highly available and durable. EBS volume data is replicated across multiple servers in an AZ to prevent the loss of data from the failure of any single component. Due to this built in protection, you don’t have to configure RAID 1 for volumes containing database transaction log files and Db2 binaries.

We don’t recommend RAID 5 for container files for data, index, or temporary tablespaces on AWS for the following reasons:

- As mentioned previously, volumes are replicated within AZ by default.
- Parity write operations of RAID 5 consume some of the Input/Output Operations Per Second (IOPS) available to your volume and will reduce the overall Input/Output (IO) available for database operations by about 20-30% over RAID 0 configuration.

Network

Ensure that you have your network constructs set up to deploy resources related to SAP NetWeaver. If you haven’t already set up network components like Amazon VPC, subnets, and route tables, you can use AWS Quick Start for VPC to easily deploy scalable VPC architecture. See the AWS Quick Start for VPC reference deployment guide.

See the series of VPC Subnet Zoning Pattern blogs for VPC patterns that you should consider for SAP applications.

Business Continuity

We recommend that you architect your business-critical applications to be fault tolerant. Depending on your availability requirements, there are different ways to achieve this. In this section we will discuss how you can set up highly available IBM Db2 for SAP applications.

High Availability

High availability for IBM Db2 database on AWS can be configured with IBM HADR with Pacemaker:
One of the requirements for automated failover with IBM Db2 HADR on AWS is Pacemaker. Implementing a Pacemaker cluster in AWS is similar to deploying it in an on-premises setting. On AWS, you need to deploy the cluster nodes in separate subnets, and we recommend that you have these subnets in different AZs.

Figure 2 provides an overview of architecture for IBM Db2 HADR with Pacemaker on AWS. This includes the following components:

- A VPC configured with two private subnets across two AZs. This provides the network infrastructure for your IBM Db2 deployment.
- In private subnet, Linux servers are configured with Pacemaker to protect the IBM Db2 database.
- Overlay IP address (similar to a virtual IP address) that is relocatable between the primary and standby Db2 databases.

Figure 2 - IBM Db2 HADR with Pacemaker

Deployment

This section discusses high level deployment process and steps. Table 2 lists the steps in the order they should be done, and each step’s purpose.

Table 2 – Steps to set up AWS resources to deploy IBM Db2 HADR with Pacemaker for SAP NetWeaver on Amazon EC2 instances

<table>
<thead>
<tr>
<th>Activity</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1: Db2 Virtual Hostname (p. 116)</td>
<td>Decide on the virtual hostname for your Db2 database (for example, dbhadb2).</td>
</tr>
<tr>
<td>Step 2: AWS Overlay IP (p. 116)</td>
<td>Decide on the Overlay IP for the dbhadb2 name (for example, 192.168.1.90).</td>
</tr>
</tbody>
</table>
**Step 1: Db2 Virtual Hostname**

Decide on the virtual hostname for your Db2 database. For example, if your virtual hostname is `dbhadb2`, it would be configured in the SAP and `dbclient` profiles. See **Step 7: Post Setup Configuration (p. 135)** in this document for more information.

**Step 2: AWS Overlay IP**

Decide on IP address to use for your Overlay IP. An Overlay IP address is an AWS-specific routing entry which can send network traffic to an instance, no matter which AZ the instance is located in.

One key requirement for the Overlay IP is that it should not be used elsewhere in your VPC or on-premises. It should be part of the private IP address range defined in RFC1918. For example, if your VPC is configured in the range of `0.0.0.0/8` or `172.16.0.0/12`, you can use the Overlay IP from the range of `192.168.0.0/16`. Based on the number of HA setups you plan to have in your landscape, you can reserve the IP address by reserving a block from the private IP address to ensure there is no overlap.

AWS worked on creating a resource agent, `aws-vpc-move-ip`, which is available along with the Linux Pacemaker. This agent updates the route table of the VPC where you have configured the cluster to always point to the primary DB.

All traffic within the VPC can reach the Overlay IP address via the route table. Traffic from outside the VPC, whether that is from another VPC or on-premises will require AWS Transit Gateway (TGW) or AWS NLB to reach the Overlay IP address. For more information on how to direct traffic to an Overlay IP address via AWS TGW or AWS NLB, see [SAP on AWS High Availability with Overlay IP Address Routing](https://help.sap.com/).

**Step 3: AWS Resources**

Deciding the right storage layout is important to ensure you can meet required IO. EBS gp2 volumes balance price and performance for a wide variety of workloads, while io1 volumes provide the highest performance consistently for mission-critical applications. With these two options, you can choose a storage solution that meets your performance and cost requirements. For more information, see [Amazon EBS features](https://aws.amazon.com/ebs/features/) for more information.

**Step 4: SAP Netweaver and IBM Db2 Deployment**

See the [SAP Standard Installation guide](https://help.sap.com/) based on your installation release to get the technical steps and prerequisites for SAP installation.
Step 4a: Create EC2 Instances for Deploying SAP NetWeaver ASCS

See SAP NetWeaver Environment Setup for Linux on AWS to learn how to set up an EC2 instance for SAP NetWeaver.

Step 4b: Create EC2 Instances for IBM Db2 Primary and Standby Databases

Deploy two EC2 instances, one in each AZ, for your primary and standby databases.

Step 4c: Disable Source/destination Check for the EC2 Instance Hosting the IBM Db2 Primary and Standby Databases

You need to disable source/destination check for your EC2 instance hosting primary and standby databases. This is required to route traffic via Overlay IP. See Changing the source or destination checking to learn more about how to disable source/destination check for your EC2 instance. You can use the following command line interface (CLI) to achieve this.

```bash
# aws ec2 modify-instance-attribute --profile cluster --instance-id EC2-instance-id --no-source-dest-check
```

Step 4d: AWS IAM Role

For the Pacemaker setup, create two policies and attach them to the IAM role, which is attached to the Db2 primary and standby instance. This allows your EC2 instance to call the APIs which run during the failover process by Pacemaker.

- STONITH – Allows Ec2 instance to start, stop and reboot instances.

```
{
    "Version": "2012-10-17",
    "Statement": [ 
    {
        "Sid": "Stmt1424870324000",
        "Effect": "Allow",
        "Action": [
            "ec2:DescribeInstances",
            "ec2:DescribeInstanceAttribute",
            "ec2:DescribeTags"
        ],
        "Resource": "*"
    },
    {
        "Sid": "Stmt1424870324001",
        "Effect": "Allow",
        "Action": [
            "ec2:ModifyInstanceAttribute",
            "ec2:RebootInstances",
            "ec2:StartInstances",
            "ec2:StopInstances"
        ],
        "Resource": [ 
```

117
Overlay IP – Allows the Ec2 instance to update the route table in case of failover.

Replace the following variables with the appropriate names:

- region-name: the name of the AWS region.
- account-id: The name of the AWS account in which the policy is used.
- rtb-XYZ: The identifier of the routing table which needs to be updated.
- i-node1: Instance ID for the Db2 primary instance.
- i-node2: Instance id for the Db2 standby instance.

Step 4e: SAP Application and Db2 Software Install

Prerequisites:

- Before starting the installation, update the /etc/hosts files of database, ASCS, and application servers with the hostname and IP address of your database, ASCS and application servers. This ensures that all your instances can resolve each other’s address during installation and configuration.
- You need to install the following packages in your instances: tcsh.x86_64, ksh.x86_64, libaio.x86_64, libstdc++.x86_64.
- Comment out the 5912 port entry in the /etc/services file (if it exists), as this port is used for the Db2 installation:
SAP application and Db2 software installation (high-level instructions):

1. Install SAP ASCS using software provisioning manager (SWPM) on the Amazon EC2 instance. Choose the installation option depending on the scenario; for example, distributed or HA in case you plan to install ERS for app layer high availability.

   ![Figure 3 – Install ASCS](image3.png)

2. Install the primary database using SWPM on the Amazon EC2 instance hosted in AZ1.

   ![Figure 4 – Install the primary database](image4.png)

3. Take a backup of the primary database.

4. Install the PAS instance. This can be the same EC2 instance used in step 1 (p. 116) if you want to install ASCS and PAS on one host.

   ![Figure 5 – Install the PAS instance](image5.png)
5. For the standby DB installation:

a. Use the SAP homogeneous system copy procedure from SWPM with the option of **System copy > Target systems > Distributed > Database instance**.

b. In the SWPM parameter screen, when asked for system load type, choose Homogenous System and the backup/restore option.

c. When prompted by the SWPM, restore the backup you took during step 3 (p. 116) on the standby DB. You can exit the installer, because the subsequent installation is already completed on the primary database server.

You should now have your ASCS, primary Db2 database, and PAS (if running on a different host than ASCS) installed and running in AZ1 of your setup. In AZ2, you should have the standby Db2 database installed and running. You can optionally install an additional application server if required to support the workload. We recommend that you distribute your application servers in both AZs.

### Step 5: Db2 HADR Setup

The following steps explain how to set up HADR between the primary and standby databases. For additional references, see:

- IBM Db2 HADR documentation
- IBM Support Page

Table 3 details the steps for setup. Update the configuration commands according to your environment.

#### Table 3 – Db2 HADR setup

<table>
<thead>
<tr>
<th>System ID (SID)</th>
<th>STJ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Db2 database hostname</td>
<td>dbprim00</td>
</tr>
<tr>
<td>Standby Db2 database hostname</td>
<td>dbsec00</td>
</tr>
<tr>
<td>Overlay IP</td>
<td>192.168.1.81</td>
</tr>
</tbody>
</table>

To set up Db2 HADR:

1. Find the state of the primary database before HADR configuration by executing the following command:

```
> db2 get db cfg for STJ | grep HADR
HADR database role             = STANDARD
HADR local host name          = (HADR_LOCAL_HOST) =
HADR local service name       = (HADR_LOCAL_SVC) =
HADR remote host name         = (HADR_REMOTE_HOST) =
HADR remote service name      = (HADR_REMOTE_SVC) =
HADR instance name of remote server = (HADR_REMOTE_INST) =
HADR timeout value            = (HADR_TIMEOUT) = 120
HADR target list              = (HADR_TARGET_LIST) =
HADR log write synchronization mode = (HADR_SYNCMODE) = NEARSYNC
HADR spool log data limit (4KB)  = (HADR_SPOOL_LIMIT) = AUTOMATIC(0)
HADR log replay delay (seconds)  = (HADR_REPLAY_DELAY) = 0
HADR peer window duration (seconds) = (HADR_PEER_WINDOW) = 0
```
2. The `HADR_LOCAL_SVC` and `HADR_REMOTE_SVC` parameters require an entry in your `/etc/services` file. If the entry is unavailable, update the `/etc/services` file to include the entry. Here is a sample `/etc/services` file entry. The SID is STJ.

```
# grep -i hadr /etc/services
STJ_HADR_1   5950/tcp   # DB2 HADR log shipping
STJ_HADR_2   5951/tcp   # DB2 HADR log shipping
```

3. Complete the following steps in your primary database (in this case, `dbprim00`) as the Db2 instance owner (in this case, `db2stj`):

```
db2 update db cfg for STJ using HADR_LOCAL_HOST dbprim00
db2 update db cfg for STJ using HADR_LOCAL_SVC STJ_HADR_1
db2 update db cfg for STJ using HADR_REMOTE_HOST dbsec00
```

Here is the state after the configuration was updated:

```
> db2 get db cfg for STJ | grep HADR
HADR database role                                      = STANDARD
HADR local host name                  (HADR_LOCAL_HOST) = dbprim00
HADR local service name                (HADR_LOCAL_SVC) = STJ_HADR_1
HADR remote host name                (HADR_REMOTE_HOST) = dbsec00
HADR remote service name              (HADR_REMOTE_SVC) = STJ_HADR_2
HADR instance name of remote server  (HADR_REMOTE_INST) = db2stj
HADR timeout value                       (HADR_TIMEOUT) = 120
HADR target list                     (HADR_TARGET_LIST) =
HADR log write synchronization mode     (HADR_SYNCMODE) = NEARSYNC
HADR spool log data limit (4KB)        (HADR_SPOOL_LIMIT) = AUTOMATIC(0)
HADR log replay delay (seconds)        (HADR_REPLAY_DELAY) = 0
HADR peer window duration (seconds)    (HADR_PEER_WINDOW) = 300
HADR SSL certificate label             (HADR_SSL_LABEL) =
```

4. Run the following steps in your standby database (in this case `dbsec00`) as the Db2 instance owner (in this case, `db2stj`).

```
db2 update db cfg for STJ using HADR_LOCAL_HOST dbsec00
db2 update db cfg for STJ using HADR_LOCAL_SVC STJ_HADR_2
```

---

121
Here's an example configuration:

```bash
> db2 get db cfg for STJ | grep HADR
HADR database role                                      = STANDBY
HADR local host name                  (HADR_LOCAL_HOST) = dbsec00
HADR local service name                (HADR_LOCAL_SVC) = STJ_HADR_2
HADR remote host name                (HADR_REMOTE_HOST) = dbprim00
HADR remote service name               (HADR_REMOTE_SVC) = STJ_HADR_1
HADR instance name of remote server   (HADR_REMOTE_INST) = db2stj
HADR timeout value                       (HADR_TIMEOUT) = 120
HADR target list                        (HADR_TARGET_LIST) =
HADR log write synchronization mode     (HADR_SYNCMODE) = NEARSYNC
HADR spool log data limit (4KB)         (HADR_SPOOL_LIMIT) = AUTOMATIC(1200000)
HADR log replay delay (seconds)         (HADR_REPLAY_DELAY) = 0
HADR peer window duration (seconds)     (HADR_PEER_WINDOW) = 300
HADR SSL certificate label             (HADR_SSL_LABEL) =
```

5. When using Linux pacemaker, use the following Db2 HADR parameters:
   - HADR peer window duration (seconds) (HADR_PEER_WINDOW) = 300
   - HADR timeout value (HADR_TIMEOUT) = 60

   We recommend that you tune these parameters after testing the failover and takeover functionality. Because individual configuration can vary, the parameter might need adjustment.

6. After your primary and standby databases have been configured, start HADR on the standby server as the HADR standby.

   ```bash
db2 start hadr on database STJ as standby
   ```

7. Start HADR on the primary database.

   ```bash
db2 start hadr on database STJ as primary
db200000i  The START HADR ON DATABASE command completed successfully.
db2pd -hadr -db STJ | head -20
Database Member 0 -- Database STJ -- Active --
   ```

   ```bash
   HADR_ROLE = PRIMARY
   REPLAY_TYPE = PHYSICAL
   HADR_SYNCMODE = NEARSYNC
   STANDBY_ID = 1
   LOG_STREAM_ID = 0
   HADR_STATE = PEER
   HADR_FLAGS = TCP_PROTOCOL
   PRIMARY_MEMBER_HOST = dbprim00
   PRIMARY_INSTANCE = db2stj
   ...
   HADR_CONNECT_STATUS = CONNECTED
   ```
Step 6: Pacemaker Cluster Setup

In this section we'll discuss the cluster setup using Linux Pacemaker on both RHEL and SLES OS. The Linux Pacemaker works as a failover Orchestrator. It monitors both the primary and standby databases, and in the event of primary database server failure it initiates an automatic HADR takeover by the standby server. The resource agents this configuration uses are as following:

- STONITH resource agent for fencing.
- The db2 database resource, which is configured in a Primary/Standby configuration.
- The aws-vpc-move-ip resource, which is built by the AWS team to handle the overlay IP switch from the Db2 primary instance to standby in the event of failure.

As mentioned in the Operating System (p. 113) section of this document, you need the correct subscription to download these resource agents.

**Important**: Change the shell environment for the `db2<sid>` user to `/bin/ksh`.

**To change the shell environment:**

1. Shut down both the database servers using `db2stop` while logged in as `db2<sid>`.
2. Install Kornshell (ksh) (if it's not already installed).
3. Run `sudo usermod -s /bin/ksh db2<sid>`.

Step 6a. Setup on RHEL

This section focuses on setting up the Pacemaker cluster on the RHEL operating system.

**To set up the pacemaker cluster on RHEL:**

1. Basic cluster configuration – Install the required cluster packages using both database nodes.

   ```
   yum install -y pcs pacemaker fence-agents-aws
   yum install -y resource-agents
   ```

2. Start the cluster services.

   ```
   systemctl start pcsd.service
   systemctl enable pcsd.service
   ```

   **Note:** If you have subscribed to RHEL for SAP with HA and US products from AWS Marketplace, run `mkdir -p /var/log/pcsd /var/log/cluster` before starting `pcsd.service`.

3. Reset the password for user hacluster on both the DB nodes.

   ```
   passwd hacluster
   ```

4. Authorize the cluster. Make sure that both nodes are able to communicate with each other using the hostname.
5. Create the cluster.

```bash
[root@dbprim00 ~]# pcs cluster auth dbprim00 dbsec00
Username: hacluster
Password:
dbprim00: Authorized
dbsec00: Authorized
[root@dbprim00 ~]#

[root@dbprim00 ~]# pcs cluster setup --name db2ha dbprim00 dbsec00
Destroying cluster on nodes: dbprim00, dbsec00...
dbsec00: Stopping Cluster (pacemaker)...
dbprim00: Stopping Cluster (pacemaker)...
dbprim00: Successfully destroyed cluster
dbsec00: Successfully destroyed cluster
Sending 'pacemaker_remote authkey' to 'dbprim00', 'dbsec00'
dbprim00: successful distribution of the file 'pacemaker_remote authkey'
dbsec00: successful distribution of the file 'pacemaker_remote authkey'
Sending cluster config files to the nodes... 
dbprim00: Succeeded
dbsec00: Succeeded
Synchronizing pcsd certificates on nodes dbprim00, dbsec00...
dbprim00: Success
dbsec00: Success
Restarting pcsd on the nodes in order to reload the certificates...
dbprim00: Success
dbsec00: Success
[root@dbprim00 ~]# pcs cluster enable --all
dbprim00: Cluster Enabled
dbsec00: Cluster Enabled
[root@dbprim00 ~]# pcs cluster start --all
dbsec00: Starting Cluster...
dbprim00: Starting Cluster...
[root@dbprim00 ~]#
```

**Note:** Adjust the corosync timeout.

6. Go to `/etc/corosync/corosync.conf` and add or modify the token value of totem to `30000`.

```bash
[root@dbprim00 corosync]# more /etc/corosync/corosync.conf
totem {
    version: 2
    cluster_name: db2ha
    secauth: off
    transport: udpu
    token: 30000
}
nodelist {
    node {
        ring0_addr: dbprim00
        nodeid: 1
    }

    node {
        ring0_addr: dbsec00
    }

```

124
nodeid: 2
}
}
}
quorum {
  provider: corosync_votequorum
  two_node: 1
}
logging {
  to_logfile: yes
  logfile: /var/log/cluster/corosync.log
  to_syslog: yes
}

7. Run `pcs cluster sync` to sync the changes on the standby database node.

```
[root@dbprim00 corosync]# pcs cluster sync
dbprim00: Succeeded
dbsec00: Succeeded
```

8. Run `pcs cluster reload corosync` to make the changes effective.

```
[root@dbprim00 corosync]# pcs cluster reload corosync
Corosync reloaded
```

9. To ensure of the changes are in place, run `corosync-cmapctl | grep totem.token`.

```
[root@dbprim00 corosync]# corosync-cmapctl | grep totem.token
runtime.config.totem.token (u32) = 30000
runtime.config.totem.token_retransmit (u32) = 7142
runtime.config.totem.token_retransmits_before_loss_const (u32) = 4
totem.token (u32) = 30000
```

10. Before creating any resource, put the cluster in maintenance mode.

```
[root@dbprim00 ~]# pcs property set maintenance-mode='true'
```

11. Create the STONITH resource. You will need the EC2 instance IDs for this operation. The default pcmk action is reboot. Replace the instance ID for `dbprim00` and `dbsec00` with the instance IDs of your setup.

If you want to have the instance remain in a stopped state until it has been investigated and then manually started, add `pcmk_reboot_action=off`. This setting is also required if you are running the Db2 on Amazon EC2 Dedicated Hosts.

```
[root@dbprim00 ~]# pcs stonith create clusterfence fence_aws
region=us-east-1 pcmk_host_map="dbprim00:i-09d1b1f05f71e5ed;dbsec00:i-00d3444601b1d8c5" power_timeout=240 pcmk_reboot_timeout=480 pcmk_reboot_retries=4
op start timeout=300
op monitor timeout=60
```
12. Create the Db2 resource.

```
[root@dbprim00 ~]# pcs resource create Db2_HADR_STJ db2
instance=db2stj dblist=STJ master meta notify=true resource-stickiness=5000 op demote timeout=240 op promote timeout=240 op start timeout=240 op stop timeout=240 op monitor interval=22s role=Master timeout=120s
```

**Note:** The timeout values here are default, which works for most deployments. We recommend that you test the timeouts in the QA setup extensively based on the test cases mentioned in the Appendix, and then tune it accordingly.

13. Create the Overlay IP resource agent. First, add the Overlay IP in the primary node.

```
[root@dbprim00 ~]# ip address add 192.168.1.81/32 dev eth0
[root@dbprim00 ~]# ip addr show
```

14. Update the route table with the Overlay IP pointing to the Db2 primary instance:

```
aws ec2 create-route --route-table-id rtb-xxxxxxxx --destination-cidr-block Overlay IP --instance-id i-xxxxxxxx
[root@dbprim00 ~]# aws ec2 create-route --route-table-id rtb-dbe0eba1 --destination-cidr-block 192.168.1.81/32 --instance-id i-09d1b1f105f71e5ed
{
  "Return": true
}
[root@dbprim00 ~]# pcs resource create db2-oip aws-vpc-move-ip
ip=192.168.1.81 interface=eth0 routing_table=rtb-dbe0eba1
```

**Note:** If you are using a different route table for both the subnets where you are deploying the primary and standby databases, you can specify them using a comma (,) in the resource creation command:

```
pcs resource create db2-oip aws-vpc-move-ip
ip=192.168.1.81 interface=eth0 routing_table=rtb-xxxxx1,rtb-xxxxx2
```

15. Create a colocation constraint to bind the Overlay IP resource agent with the Db2 primary instance.
Step 6a. Setup on RHEL

```bash
[root@dbprim00 ~]# pcs constraint colocation add db2-oip with master Db2_HADR_STJ-master 2000
```

You can now remove the maintenance mode by using the following code:

```bash
[root@dbprim00 ~]# pcs property set maintenance-mode='false'
```

This is the final configuration of the cluster:

```bash
[root@dbprim00 ~]# pcs config show
Cluster Name: db2ha
Corosync Nodes:
  dbprim00 dbsec00
Pacemaker Nodes:
  dbprim00 dbsec00

Resources:
  Master: Db2_HADR_STJ-master
  Resource: Db2_HADR_STJ (class=ocf provider=heartbeat type=db2)
    Attributes: dblist=STJ instance=db2stj
    Meta Attrs: notify=true resource-stickiness=5000
    Operations: demote interval=0s timeout=120 (Db2_HADR_STJ-demote-interval-0s)
    monitor interval=20 timeout=60 (Db2_HADR_STJ-monitor-interval-20)
    notify interval=0s timeout=10 (Db2_HADR_STJ-notify-interval-0s)
    promote interval=0s timeout=120 (Db2_HADR_STJ-promote-interval-0s)
    start interval=0s timeout=120 (Db2_HADR_STJ-start-interval-0s)
    stop interval=0s timeout=120 (Db2_HADR_STJ-stop-interval-0s)
  Resource: db2-oip (class=ocf provider=heartbeat type=aws-vpc-move-ip)
    Attributes: interface=eth0 ip=192.168.1.81 routing_table=rtb-dbe0eba1
    Operations: monitor interval=60 timeout=30 (db2-oip-monitor-interval-60)
    start interval=0s timeout=180 (db2-oip-start-interval-0s)
    stop interval=0s timeout=180 (db2-oip-stop-interval-0s)

Stonith Devices:
  Resource: clusterfence (class=stonith type=fence_aws)
    Attributes:
    pcmk_host_map=dbprim00:1-09d1bf105f71e5ed;dbsec00:1-0c0d344601bd8c5
    pcmk_reboot_retries=4 pcmk_reboot_timeout=480 power_timeout=240 region=us-east-1
    Operations: monitor interval=60s (clusterfence-monitor-interval-60s)

Fencing Levels:
  Location Constraints:
  Ordering Constraints:
  Colocation Constraints:
    db2-oip with Db2_HADR_STJ-master (score:2000) (rsc-role:Started) (with-rsc-role:Master)
  Ticket Constraints:

Alerts:
  No alerts defined

Resources Defaults:
  No defaults set

Operations Defaults:
  No defaults set
```
Step 6b. Setup on SLES

This section focuses on setting up the Pacemaker cluster on the SLES operating system.

**Prerequisite:** You need to complete this on both the Db2 primary and standby instances.

**To create an AWS CLI profile:**

The SLES operating system's resource agents use the AWS Command Line Interface (CLI). You need to create the AWS CLI profile for the root account on both instances: one with the default profile and the other with an arbitrary profile name (in this example, `cluster`) which creates output in text format. The region of the instance must be added as well.

1. Replace the string `region-name` with your target region in the following example.

   ```bash
   dbprim00:~ # aws configure
   AWS Access Key ID [None]:
   AWS Secret Access Key [None]:
   Default region name [None]: us-east-1
   Default output format [None]:
   dbprim00:~ # aws configure --profile cluster
   AWS Access Key ID [None]:
   AWS Secret Access Key [None]:
   Default region name [None]: us-east-1
   Default output format [None]: text
   ```

   You don’t need to provide the Access Key and Secret Access key, because access is controlled by the IAM role you created earlier in the setup.

2. Add a second private IP address.

3. You are required to add a second private IP address for each cluster instance. Adding a second IP address to the instance allows the SUSE cluster to implement a two-ring corosync configuration. The two-ring corosync configuration allows the cluster nodes to communicate with each other using the secondary IP address if there is an issue communicating with each other over the primary IP address. See [To assign a secondary private IPv4 address to a network interface](#).

4. Add a tag with an arbitrary “key” (in this case, `pacemaker`). The value of this tag is the hostname of the respective Db2 instance. This is required to enable AWS CLI to use filters in the API calls.

5. Disable the source/destination check.

6. Ensure that the source/destination check is disabled, as described in [Step 4c (p. 117)](#).

7. Avoid deletion of cluster-managed IP addresses on the `eth0` interface.

8. Check if the package `cloud-netconfig-ec2` is installed with the following command:
9. Update the file `/etc/sysconfig/network/ifcfg-eth0` if this package is installed. Change the following line to a 'no' setting or add the following line if the package is not yet installed:

```bash
dbprim00:~ # CLOUD_NETCONFIG_MANAGE='no'
```

10. Set up NTP (best with YaST). Use AWS time service at 169.254.169.123, which is accessible from all EC2 instances. Enable ongoing synchronization.

11. Activate the public cloud module to get updates for the AWS CLI:

```bash
dbprim00:~ # SUSEConnect --list-extensions
dbprim00:~ # SUSEConnect -p sle-module-public-cloud/12/x86_64
Registering system to registration proxy https://smt-ec2.susecloud.net
Updating system details on https://smt-ec2.susecloud.net ...
Activating sle-module-public-cloud 12 x86_64 ...
-> Adding service to system ...
-> Installing release package ...
Successfully registered system
```

12. Update your packages on both the with the command:

```bash
dbprim00:~ # zypper -n update
```

13. Install the resource agent pattern `ha_sles`.

```bash
dbprim00:~ # zypper install -t pattern ha_sles
```

To configure pacemaker: Configuration of the `corosync.conf` file:

1. Use the following configuration in the `/etc/corosync/corosync.conf` file on both the Db2 primary and standby instances:

```bash
# Read the corosync.conf.5 manual page
totem {
  version: 2
  rrp_mode: passive
  token: 30000
  consensus: 36000
  token_retransmits_before_loss_const: 10
  max_messages: 20
  crypto_cipher: none
  crypto_hash: none
  clear_node_high_bit: yes
  interface {
    ringnumber: 0
    bindnetaddr: <ip-local-node>
    mcastport: 5405
    ttl: 1
  }
  transport: udpu
}
```
Step 6b. Setup on SLES

logging {
  fileline: off
  to_logfile: yes
  to_syslog: yes
  logfile: /var/log/cluster/corosync.log
  debug: off
  timestamp: on
  logger_subsys {
    subsys: QUORUM
    debug: off
  }
}

nodelist {
  node {
    ring0_addr: <ip-node-1>
    ring1_addr: <ip2-node-1>
    nodeid: 1
  }
  node {
    ring0_addr: <ip-node-2>
    ring1_addr: <ip2-node-2>
    nodeid: 2
  }
}

quorum {
  # Enable and configure quorum subsystem (default: off)
  # see also corosync.conf.5 and votequorum.5
  provider: corosync_votequorum
  expected_votes: 2
  two_node: 1
}

2. Replace the variables `ip-node-1` / `ip2-node-1` and `ip-node-2` / `ip2-node-2` with the IP addresses of your Db2 primary and standby instances, respectively. Replace `ip-local-node` with the IP address of the instance on which the file is being created.

The chosen settings for `crypto_cipher` and `crypto_hash` are suitable for clusters in AWS. They may be modified according to SUSE's documentation if you want strong encryption of cluster communication.

3. Start the cluster services and enable them on both the Db2 primary and standby instances.

```
dbprim00:~ # systemctl start pacemaker
Created symlink from /etc/systemd/system/multi-user.target.wants/pacemaker.service to /usr/lib/systemd/system/pacemaker.service.
dbprim00:~ # systemctl enable pacemaker
```

4. Check the configuration with the following command:

```
dbprim00:~ # corosync-cfgtool -s
Printing ring status.
Local node ID 1
RING ID 0
  id   = 10.0.1.17
  status = ring 0 active with no faults
RING ID 1
  id   = 10.0.1.62
  status = ring 1 active with no faults
dbprim00:~ #
```
Cluster status:

dbprim00:~ # crm_mon -l
Stack: corosync
Current DC: dbsec00 (version 1.1.19+20181105.ccd6b5b10-3.13.1-1.1.19+20181105.ccd6b5b10)
  - partition with quorum
Last updated: Fri Apr 17 14:09:56 2020
Last change: Fri Apr 17 13:38:59 2020 by hacluster via crmd on dbsec00

2 nodes configured
0 resources configured

Online: [ dbprim00 dbsec00 ]

No active resources

To prepare the cluster for adding resources:

1. To avoid cluster starting partially defined resources, set the cluster to maintenance mode. This deactivates all monitor actions.

   dbprim00:~ # crm configure property maintenance-mode="true"
   dbprim00:~ # crm status
   Stack: corosync
   Current DC: dbprim00 (version 1.1.19+20181105.ccd6b5b10-3.16.1-1.1.19+20181105.ccd6b5b10)
   - partition with quorum
   Last updated: Fri Apr 17 14:30:51 2020
   Last change: Fri Apr 17 14:30:50 2020 by root via cibadmin on dbprim00

   2 nodes configured
   0 resources configured

   *** Resource management is DISABLED ***
   The cluster will not attempt to start, stop or recover services

   Online: [ dbprim00 dbsec00 ]

   No resources

2. Configuring AWS-specific settings:

   dbprim00:/ha-files # vi crm-bs.txt
   dbprim00:/ha-files # more crm-bs.txt
   property cib-bootstrap-options:
   "stonith-enabled="true""
   "stonith-action="off""
   "stonith-timeout="600s"
   rsc_defaults rsc-options:
   "resource-stickiness=1"
   "migration-threshold=3"
   op_defaults op-options:
   "timeout=600"
   "record-pending=true"

   The off setting forces the agents to shut down the instance. You have the option of changing it to reboot if required.
3. Add the following configuration to the cluster:

```
dbprim00:~ # crm configure load update crm-bs.txt
```

**To configure the AWS-specific STONITH resource #**:

1. Create a file with the following content:

```
primitive res_AWS_STONITH stonith:external/ec2 \
    op start interval=0 timeout=180 \ 
    op stop interval=0 timeout=180 \ 
    op monitor interval=120 timeout=60 \ 
    params tag=pacemaker profile=cluster
```

The EC2 tag `pacemaker` entry needs to match the tag chosen for the EC2 instances, and the name of the profile needs to match the previously configured AWS profile as part of the prerequisite section.

2. Add the file to the configuration:

```
dbprim00:/ha-files # vi aws-stonith.txt
```
```
dbprim00:/ha-files # crm configure load update aws-stonith.txt
```

3. Create the Db2 Primary/Standby resource.

4. Create a file with the following content. Change the value for SID, as per your configuration.

```
primitive rsc_db2_db2stj_STJ db2 \ 
    params instance="db2stj" dblist="STJ" \ 
    op start interval="0" timeout="130" \ 
    op stop interval="0" timeout="120" \ 
    op promote interval="0" timeout="120" \ 
    op demote interval="0" timeout="120" \ 
    op monitor interval="30" timeout="60" \ 
    op monitor interval="31" role="Master" timeout="60"
ms msl_db2_db2stj_STJ rsc_db2_db2stj_STJ \ 
    meta target-role="Started" notify="true"
```

5. Add the file to the configuration:

```
dbprim00:/ha-files # vi db2res.txt
```
```
dbprim00:/ha-files # crm configure load update db2res.txt
```
```
primitive rsc_db2_db2stj_STJ db2 \ 
    params instance="db2stj" dblist="STJ" \ 
    op start interval="0" timeout="130" \ 
    op stop interval="0" timeout="120"
```
6. Create the Overlay IP resource agent.

a. First, add the Overlay IP in the Db2 primary instance.

```
dbprim00:~# ip address add 192.168.1.81/32 dev eth0
```

b. Update the route table with the Overlay IP pointing to the Db2 primary instance.

```
aws ec2 create-route --route-table-id rtb-xxxxxxxx --destination-cidr-block Overlay IP --instance-id i-xxxxxxxx
```

**Note:** If you are using different route table for both the subnets where you are deploying the primary and standby database, you can specify them with a comma (,) in the command preceding this note.

```
dbprim00:~ # aws ec2 create-route --route-table-id rtb-dbe0ebal --destination-cidr-block 192.168.1.81/32 --instance-id i-05fc8801284585362

{  
  "Return": true
}
```

The `aws-vpc-move-ip` resource agent call the AWS command from the location `/usr/bin`, so ensure that there is a soft link pointing to the location where you have the `awscliv2` installed.

```
dbprim00:/usr/bin # which aws
/usr/local/bin/aws

dbprim00:/usr/bin # ls -ltr aws
lrwxrwxrwx 1 root root 18 Apr 18 17:44 aws -> /usr/local/bin/aws
```
c. Create the file with the following content, and replace the Overlay IP and the route table ID based on your configuration. If you have multiple route tables associated with the subnet to which your instances belong, you can use a comma-separated list of routing tables.

Note: Make sure you use the same profile name (which is cluster for this setup) that you used while configuring the AWS CLI.

```
dbprim00:/ha-files # vi aws-move-ip.txt
dbprim00:/ha-files # more aws-move-ip.txt
primitive res_AWS_IP ocf:suse:aws-vpc-move-ip
params ip=192.168.1.81 routing_table=rtb-dbe0ebal interface=eth0
profile=cluster
    op start interval=0 timeout=180
    op stop interval=0 timeout=180
    op monitor interval=60 timeout=60
dbprim00:/ha-files #crm configure load update aws-move-ip.txt
```

d. Create a colocation constraint to bind the Overlay IP resource agent with the Db2 primary instance.

```
dbprim00:/ha-files # more crm-cs.txt
colocation col_db2_db2stj_STJ 2000: res_AWS_IP:Started
    msl_db2_db2stj_STJ:Master
dbprim00:/ha-files #crm configure load update crm-cs.txt
dbprim00:/ha-files #
```

e. Adjust the resource-stickiness and migration-threshold values.

```
dbprim00:~ # crm configure rsc_defaults resource-stickiness=1000
dbprim00:~ # crm configure rsc_defaults migration-threshold=5000
```

f. You can now remove maintenance-mode.

```
dbprim00:~ # crm configure property maintenance-mode="false"
```

Final configuration of the cluster:

```
dbprim00:/ha-files # crm status
Stack: corosync
Current DC: dbsec00 (version
1.1.19+20181105.ccd6b5b10-3.16.1-1.1.19+20181105.ccd6b5b10) - partition with quorum
Last updated: Sat Apr 18 18:45:53 2020
Last change: Sat Apr 18 16:01:26 2020 by root via cibadmin on dbprim00

2 nodes configured
4 resources configured

Online: [ dbprim00 dbsec00 ]

Full list of resources:
```

Step 7: Post Setup Configuration

To enable SAP to connect to the Db2 virtual name, post-setup configuration tasks must be performed.

To perform post-setup configuration tasks:

1. Edit your SAP profile files:

```
> vi DEFAULT.PFL

SAPDBHOST = dbhadb2
j2ee/dbhost = dbhadb2
rsdb/reco_trials = 10
rsdb/reco_sleep_time = 10
```
2. Update the two parameters (SAPDBHOST and j2ee/dbhost) to the virtual name you chose for your database server. You will have to update the rsab/reco* parameters to greater than failover duration to avoid DB disconnect in case of failover. We recommend that you test these values in QA before setting it up in production.

3. Edit your Db2 client file:

```plaintext
> cd /sapmnt/STJ/global/db6
sappas01:stjadm 15> more db2cli.ini
; Comment lines start with a semi-colon.
[STJ]
  Database=STJ
  Protocol=tcpip
  Hostname=dbhadb2
  Servicename=5912
[COMMON]
  Diagpath=/usr/sap/STJ/SYS/global/db6/db2dump
```

Make sure the hostname parameter matches your Db2 virtual hostname.

4. After you change the entries and save your file, test your connection to the database server:

```plaintext
sappas01:stjadm 17> R3trans -d
This is R3trans version 6.26 (release 745 - 13.04.18 - 20:18:04).
unicode enabled version
R3trans finished (0000).
sappas01:stjadm 18> startsap
Checking db Database
Database is running
Starting Startup Agent sapstartsrv
OK
Instance Service on host sappas01 started
starting SAP Instance D00
Startup-Log is written to /home/stjadm/startsap_D00.log
/usr/sap/STJ/D00/exe/sapcontrol -prot NI_HTTP -nr 00 -function Start
Instance on host sappas01 started
```

![Host data and Database data table]

Figure 6 – SAP system status information

You can check get the status/information of HADR in the transaction DB02/dbacockpit > Configuration > Overview.
Step 8: Testing and Validation

We recommend you define your failure scenarios and test them on your cluster. Unless otherwise specified, all tests are done with the primary node running on the primary server (dbprim00) and the standby node running on the standby server (dbsec00).

**Prerequisite:** Before running any tests, please ensure that:

- There is no error or failed action in the Pacemaker. This can be tested using pcs status. In case there is any failed action, check the cause in /var/log/cluster/corosync.log in the node on which it
has failed, and then take the corrective action. You can clean the failed action using `pcs/crm resource cleanup`.

- There is no unintended location constraint set up. Using the `pcs/crm` resource, move the master from primary to standby to set a location constraint on the primary node which prevents any resource from starting on it. This can be identified using the `pcs/crm constraint show`. Note the ID of the location constraint, and then run `pcs/crm constraint delete <id>` to remove it.
- The Db2 HADR synchronization is working. This can be checked using `db2pd -hadr -db <DBSID>` and comparing the `LOG_FILE`, `PAGE`, and `POS` for primary and standby.
- Refer to Appendix 1 (p. 141) for detailed test cases on RHEL setup.
- Refer to Appendix 2 (p. 147) for detailed test cases on SLES Setup

**Operations**

In this section we will cover some of the native AWS services that help you with day-to-day operations of your IBM Db2 database for SAP applications.

**Monitoring**

AWS provides multiple native services to monitor and manage your infrastructure and applications on AWS. Services like Amazon CloudWatch and AWS CloudTrail can be leveraged to monitor your underlying infrastructure and APIs, respectively.

CloudWatch provides ready-to-use key performance indicators (KPIs) that you can use to monitor CPU utilization and disk utilization.

You can also create custom metrics for monitoring IBM Db2.

With AWS CloudTrail, you can log, continuously monitor, and retain account activity related to actions across your AWS infrastructure. AWS CloudTrail is enabled on all AWS accounts, and records your account activity upon account creation. You can view and download the last 90 days of your account activity for create, modify, and delete operations of supported services without the need to manually set up CloudTrail.

**Backup and Recovery**

You need to regularly back up your operating system and database to recover them in case of failure. AWS provides various services and tools that you can use to back up your IBM Db2 database of SAP applications.

**AWS Backup**

AWS Backup is a fully managed backup service centralizes and automates the backup of data across AWS services. Using AWS Backup, you can centrally configure backup policies and monitor backup activity for AWS resources, such as EBS volumes, Amazon EC2 instances, and Amazon Elastic File Systems (Amazon EFS). AWS Backup automates and consolidates backup tasks previously performed service-by-service, removing the need to create custom scripts and manual processes. AWS Backup provides a fully managed, policy-based backup solution, simplifying your backup management and enabling you to meet your business and regulatory backup compliance requirements.

**Amazon AMI**

You can use the AWS Management Console or the AWS CLI to create a new Amazon Machine Image (Amazon AMI) of your existing SAP system. This can be used to recover your existing SAP system or create a clone.
The AWS CLI create image command creates a new AMI based on an existing Amazon EC2 instance. The new AMI contains a complete copy of the operating system and its configuration, software configurations, and optionally all EBS volumes that are attached to the instance.

A simple command to create an AMI with reboot (if running) of your EC2 instance (with instance ID i-0b09a25c58929de26 as example) including all attached EBS volumes:

```
aws ec2 create-image --instance-id i-0b09a25c58929de26 --name "My server"
```

A simple command to create AMI without reboot (if running) of your EC2 instance (with instance ID i-0b09a25c58929de26 as example) including all attached EBS volumes:

```
aws ec2 create-image --instance-id i-0b09a25c58929de26 --name "My server" --no-reboot
```

**Amazon EBS Snapshots**

You can back up your Amazon EBS volumes to Amazon S3 by taking point-in-time snapshots. Snapshots are incremental backups, which means that only the blocks on the device that have changed after your most recent snapshot are saved.

Snapshots are suited to backup SAP file systems like `/usr/sap/*`, `/sapmnt/*`. We do not recommend using snapshots to back up your volumes containing data and log files. If you decide to take snapshots for your database volume snapshot, keep in mind that for consistency you should use Microsoft’s Volume Shadow Copy Service and use the `run command` to back up or shut down your database before Snapshots is triggered.

A simple command to create a snapshot of volume (with volume id `vol-1234567890abcdef0` as example):

```
aws ec2 create-snapshot --volume-id vol-1234567890abcdef0 --description "This is my volume snapshot."
```

**Database Backups**

One of following methods can be used for IBM Db2 database backup:

- **With native tools to take backup on disk** — Backup requires high throughput compared to Input/Output Operations Per Second (IOPS). We recommend using st1 disk, which provides maximum throughput of 500MB/s per volume. Once the backup completes on disk it can be moved to an Amazon S3 bucket via scripts.

- **With third party backint tools** — There are many third-party tools from partners like Commvault and Veritas that use SAP backint interface and store backups directly in Amazon S3 buckets.

**Storage**

The storage services we use across this guide are:
Amazon EBS

Amazon EBS provides persistent storage for SAP applications and databases. EBS volume size can be increased or their type can be changed (for example, gp2 to io1) without downtime requirements. For more information, see Modifying Amazon EBS volume.

Once you have extended the volume, you need to extend the drive with your Linux volume manager software.

Amazon S3

Amazon S3 does not need you to explicitly provision storage at all – you simply pay for what you use.

Operating System Maintenance

Operating system maintenance across large estates of EC2 instances can be managed by:

- Tools specific to each operating system such as SUSE Manager and Red Hat Smart Management.
- 3rd party products such as those available on the AWS Marketplace.
- Using AWS Systems Manager.

Here are some key operating system maintenance tasks that AWS Systems Manager can help with:

Patching

Follow SAP recommended patching processes to update your landscape on AWS. For operating system patching, with AWS Systems Manager Patch Manager you can roll out OS patches as per your corporate policies. There are multiple key features such as:

- Scheduling based on tags
- Auto-approving patches with lists of approved and rejected patches
- Defining patch baselines

AWS Systems Manager Patch Manager integrates with AWS IAM, AWS CloudTrail, and Amazon CloudWatch Events to provide a secure patching experience that includes event notifications and the ability to audit usage. For details about the process, see How Patch Manager Operations Work. If AWS Systems Manager Patch Manager does not fulfil your requirements, there are third-party products available as well. Some of these are available via the AWS Marketplace.

Maintenance Window

AWS Systems Manager Maintenance Windows enables you to define a schedule for when to perform potentially disruptive actions on your instances, such as patching an operating system, updating drivers, or installing software or patches.

Automation Using Documents

AWS Systems Manager Automation simplifies common maintenance and deployment tasks of Amazon EC2 instances and other AWS resources. Automation enables you to do the following:

- Build automation workflows to configure and manage instances and AWS resources.
- Create custom workflows or use pre-defined workflows maintained by AWS.
- Receive notifications about Automation tasks and workflows by using Amazon CloudWatch Events.
- Monitor automation progress and execution details by using the Amazon EC2 or the AWS Systems Manager console.
Business Continuity

AWS recommends periodically scheduling business continuity process validations by executing disaster recovery tests. This planned activity helps to flush out any potential unknowns, and helps the organization deal with any real disaster in a streamlined manner. Depending on your disaster recovery architecture it may include:

- Backup/Recovery of databases from S3.
- Creation of systems from AMI and point-in-time recovery via snapshots.
- Changing EC2 instance size of pilot light systems.
- Validation of integration (AD/DNS, email, 3rd party, and more)

Support

SAP requires customers to have a minimum AWS Business Support plan with AWS. This ensures that any critical issues raised with SAP are also handled by AWS on priority. AWS business support provides a less than one-hour response time for production-down scenarios. You can also choose to have an AWS enterprise support plan, which provides a less than 15-minute response time for business-critical systems, along with other benefits. See AWS Enterprise Support.

For any SAP application issues, AWS suggests raising an incident with SAP via the SAP support portal. After the first level of investigation, SAP can redirect the incident to AWS support if they find an infrastructure related issue which needs to be managed by AWS. However, if you choose to raise support issues for SAP applications with AWS support, we cannot redirect the tickets to SAP. For any infrastructure related issues, you can raise the issue directly with AWS support.

Cost Optimization

Resources (CPU, memory, additional application servers, system copies for different tests/validations and more) required the SAP landscape change over time. AWS recommends monitoring system utilization, and the need for existing systems, on a regular basis to take actions that will reduce cost. In cases of databases like IBM Db2 as we cannot scale out only opportunity to right size database server is by scaling up/down or shutting it down if not required. A few suggestions to consider:

- Consider reserved instances or savings plans over on-demand instances if your requirement is to run 24-7, 365 days a year. Reserved instances provide up to 75% discount over on-demand instances. See Amazon EC2 pricing.
- Consider running occasionally required systems like training and sandbox on-demand for the duration required.
- Monitor CPU and memory utilization overtime for other non-production systems like Dev/QA, and right-size them when possible.

Appendix 1: Testing on RHEL Setup

Test Case 1: Manual Failover

Procedure: Use the command pcs resource move <Db2 master resource name>.

```
[root@dbprim00 profile]# pcs resource move Db2_HADR_STJ-master
Warning: Creating location constraint cli-ban-Db2_HADR_STJ-master-on-dbprim00 with a score of -INFINITY for resource
```
Test Case 1: Manual Failover

Db2_HADR_STJ-master on-dbprim00 with a score of -INFINITY for resource Db2_HADR_STJ-master on node dbprim00. This will prevent Db2_HADR_STJ-master from running on dbprim00 until the constraint is removed. This will be the case even if dbprim00 is the last node in the cluster.

[root@dbprim00 profile]#

Expected result: The Db2 primary node is moved from primary node to standby node.

[root@dbprim00 profile]# pcs status
Cluster name: db2ha
Stack: corosync
Current DC: dbsec00 (version 1.1.18-11.el7_5.4-2b07d59a9) - partition with quorum
Last updated: Sat Feb 8 08:54:04 2020
Last change: Sat Feb 8 08:53:02 2020 by root via crm_resource on dbprim00

2 nodes configured
4 resources configured
Online: [ dbprim00 dbsec00 ]

Full list of resources:

clusterfence   (stonith:fence_aws):    Started dbprim00
    Master/Slave Set: Db2_HADR_STJ-master [Db2_HADR_STJ]
    Masters: [ dbsec00 ]
    Stopped: [ dbprim00 ]
db2-oip        (ocf::heartbeat:aws-vpc-move-ip):       Started dbsec00

Daemon Status:
corosync: active/enabled
pacemaker: active/enabled
pcsd: active/enabled

[root@dbprim00 profile]#

Followup actions: Remove the location constraint.

When using a manual command for moving the resource, there is location constraint created on the node (in this case, the primary node) that prevents running the Db2 resource in standby mode.

To remove the location constraint:

1. Use the following command to remove the location constraint:

   # pcs config show
   Location Constraints:
   Resource: Db2_HADR_STJ-master
   Disabled on: dbprim00 (score:-INFINITY) (role: Started) (id:cli-ban-Db2_HADR_STJ-master-on-dbprim00)

   [root@dbprim00 profile]# pcs constraint delete cli-ban-Db2_HADR_STJ-master-on-dbprim00

2. Start the Db2 instance as standby on the new standby node, logged in as db2<sid>. Next, clean up the error logged in as root.

   db2stj> db2start
Test Case 2: Shut Down the Primary EC2 Instance

Procedure: Using AWS Console or CLI to stop the EC2 instance and simulate EC2 failure.

Expected result: The Db2 primary node is moved to the standby server.
Test Case 3: Stop the Db2 Instance on the Primary Instance

**Procedure:** Log in to the Db2 primary instance as `db2<sid>` (db2stj) and run `db2stop force`.

```
db2stj> db2stop force
02/12/2020 12:40:03 0 0 SQL1064N DB2STOP processing was successful.
```

**Expected result:** The Db2 primary node is failed over to standby server. The standby node continues to be on the old primary in a stopped state. There is a failed monitoring action.

```
[root@dbsec00 db2stj]# pcs status
Cluster name: db2ha
Stack: corosync
Current DC: dbsec00 (version 1.1.18-11.el7_5.4-2b07d5c5a9) - partition with quorum
Last updated: Wed Feb 12 16:55:56 2020
Last change: Wed Feb 12 13:58:11 2020 by hacluster via crmd on dbsec00

  2 nodes configured
  4 resources configured

    Online: [ dbprim00 dbsec00 ]

Full list of resources:

  * clustrfence   (stonith:fence_aws):    Started dbsec00
    Master/Slave Set: Db2_HADR_STJ-master [Db2_HADR_STJ]
    Masters: [ dbsec00 ]
    Stopped: [ dbprim00 ]
  * db2-oip        (ocf::heartbeat:aws-vpc-move-ip):       Started dbsec00

Failed Actions:
  * Db2_HADR_STJ_start_0 on dbprim00 'unknown error' (1): call=34, status=complete, exitreason='', last-rc-change='Wed Feb 12 16:55:32 2020', queued=1ms, exec=6749ms

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@dbsec00 db2stj]#
```

**Followup action:** Start the EC2 instance, then start Db2 as standby on the standby instance as you did in Test Case 2 (p. 143). Clear the failed monitoring error.
**Test Case 4: End the Db2 Process (db2sysc) on the Node that Runs the Primary Database**

**Procedure:** Log in to the Db2 primary instance as root and then run `ps -ef|grep db2sysc`. Note the process ID (PID) and then end it.

```
[root@dbprim00 ~]# ps -ef|grep db2sysc
root      5809 30644  0 18:54 pts/1  00:00:00 grep --color=auto
db2sysc
[output truncated]
[root@dbprim00 ~]# kill -9 26982
```

**Expected result:** The Db2 primary node is failed over to the standby server. The standby node is in the old primary in a stopped state.

```
[root@dbprim00 ~]# pcs status
Cluster name: db2ha
Stack: corosync
Current DC: dbsec00 (version 1.1.18-11.el7_5.4-2b07d5c5a9) - partition with quorum
Last updated: Wed Feb 12 18:54:50 2020
Last change: Wed Feb 12 18:53:12 2020 by hacluster via crmd on dbsec00
2 nodes configured
4 resources configured
Online: [ dbprim00 dbsec00 ]
Full list of resources:
clusterfence   (stonith:fence_aws):    Started dbsec00
Master/Slave Set: Db2_HADR_STJ-master [Db2_HADR_STJ]
   Masters: [ dbsec00 ]
   Stopped: [ dbprim00 ]
db2-oip        (ocf::heartbeat:aws-vpc-move-ip):       Started dbsec00
Failed Actions:
* Db2_HADR_STJ_start_0 on dbprim00 'unknown error' (1): call=57, status=complete,
exitreason='',
   last-rc-change='Wed Feb 12 18:54:37 2020', queued=0ms, exec=6777ms

Daemon Status:
corosync: active/enabled
pacemaker: active/enabled
pcsd: active/enabled
```

**Followup action:** Start the EC2 instance and start Db2 as standby on the standby instance, as you did in Test Case 2 (p. 143). Clear the failed monitoring alert.

**Test Case 5: End the Db2 Process (db2sysc) on the Node that Runs the Standby Database**

**Procedure:** Log in to the Db2 standby instance as root and run `ps -ef|grep db2sysc`. Note the PID and then end it.
Test Case 6: Simulating a Crash of the Node that Runs the Primary Db2

Procedure: Log in to the Db2 primary instance as root and run `echo 'c' > /proc/sysrq-trigger`.

Follow-up action: Clear the monitoring error.

Expected result: The `db2sysc` process is restarted on the Db2 standby instance. There is a monitoring failure event record in the cluster.

Follow-up action: Clear the monitoring error.
Session stopped
- Press <return> to exit tab
- Press R to restart session
- Press S to save terminal output to file

Network error: Software caused connection abort

**Expected result:** The primary Db2 should failover to standby node. The standby is in a stopped state on the previous primary.

```
[root@dbsec00 ~]# pcs status
Cluster name: db2ha
Stack: corosync
Current DC: dbsec00 (version 1.1.18-11.el7_5.4-2b07d5c5a9) - partition with quorum
Last updated: Fri Feb 21 15:38:43 2020
Last change: Fri Feb 21 15:33:17 2020 by hacluster via crmd on dbsec00

2 nodes configured
4 resources configured

Online: [ dbprim00 dbsec00 ]

Full list of resources:
```
clusterfence   (stonith:fence_aws):    Started dbsec00
Master/Slave Set: Db2_HADR_STJ-master [Db2_HADR_STJ]
      Masters: [ dbsec00 ]
      Stopped: [ dbprim00 ]

```
db2-oip        (ocf::heartbeat:aws-vpc-move-ip):       Started dbsec00
```

Failed Actions:
- Db2_HADR_STJ_start_0 on dbprim00 'unknown error' (1): call=15, status=complete,
  exitreason='',
  last-rc-change='Fri Feb 21 15:38:31 2020', queued=0ms, exec=7666ms

Daemon Status:
corosync: active/enabled
pacemaker: active/enabled
pcsd: active/enabled

**Followup action:** Start the EC2 instance and then start Db2 as standby on the standby instance as you did in Test Case 2 (p. 143). Clear the monitoring error.

---

Appendix 2: Testing on SLES Setup

**Test Case 1: Manual Failover**

**Procedure:** Use the command `crm resource move <Db2 primary resource name> force` to move the primary Db2 instance to standby node.

```
dbprim00: # crm resource move msl_db2_db2stj_STJ force
INFO: Move constraint created for rsc_db2_db2stj_STJ
```

---

147
**Expected result:** The Db2 primary node is moved from the primary node (dbprim00) to the standby node (dbsec00).

```plaintext
dbprim00:~ # crm status
Stack: corosync
Current DC: dbsec00 (version 1.1.19+20181105.ccd6b5b10-3.16.1-1.1.19+20181105.ccd6b5b10) - partition with quorum
Last updated: Sat Apr 25 19:03:20 2020
Last change: Sat Apr 25 19:02:26 2020 by root via crm_resource on dbprim00

2 nodes configured
4 resources configured

Online: [ dbprim00 dbsec00 ]

Full list of resources:

res_AWS_STONITH (stonith:external/ec2): Started dbsec00
Master/Slave Set: msl_db2_db2stj_STJ [rsc_db2_db2stj_STJ]
  Masters: [ dbsec00 ]
  Stopped: [ dbprim00 ]
res_AWS_IP (ocf::suse:aws-vpc-move-ip): Started dbsec00
```

**Follow-up actions:** Start the Db2 instance as standby on the new standby node, logged in as db2<sid>. Clean up the error logged in as root.

```plaintext
db2stj> db2start
04/25/2020 19:05:27 0 0 SQL1063N DB2START processing was successful.
SQL1063N DB2START processing was successful.

db2stj> db2 start hadr on database STJ as standby
DB20000I The START HADR ON DATABASE command completed successfully.
```

**Remove location constraint:** When using a manual command to move the resource, there is a location constraint created on the node (in this case primary node) which is run, preventing the Db2 resource from running in standby mode.

Use the following command to remove the location constraint.

```plaintext
# dbprim00: # crm resource clear msl_db2_db2stj_STJ
```

Once the constraint is removed, the standby instance starts automatically.

```plaintext
# dbprim00: # crm status
Stack: corosync
Current DC: dbsec00 (version 1.1.19+20181105.ccd6b5b10-3.16.1-1.1.19+20181105.ccd6b5b10) - partition with quorum
Last updated: Sat Apr 25 19:05:29 2020
Last change: Sat Apr 25 19:05:18 2020 by root via crm_resource on dbprim00

2 nodes configured
```
Test Case 2: Shut Down the Primary EC2 Instance

**Procedure:** Using AWS console or CLI, stop the EC2 instance to simulate EC2 failure.

**Expected Result:** The Db2 primary node is moved to a standby server (dbsec00).

```
dbsec00:~ # crm status
Stack: corosync
Current DC: dbsec00 (version 1.1.19+20181105.ccd6b5b10-3.16.1-1.1.19+20181105.ccd6b5b10) - partition with quorum
  Last updated: Sat Apr 25 19:19:32 2020
  Last change: Sat Apr 25 19:18:16 2020 by root via crm_resource on dbprim00
  2 nodes configured
  4 resources configured

Online: [ dbsec00 ]
OFFLINE: [ dbprim00 ]

Full list of resources:

  res_AWS_STONITH (stonith:external/ec2): Started dbsec00
  Master/Slave Set: msl_db2_db2stj_STJ [rsc_db2_db2stj_STJ]
    Masters: [ dbsec00 ]
    Stopped: [ dbprim00 ]
  res_AWS_IP (ocf::suse:aws-vpc-move-ip): Started dbsec00
```

**Follow-up action:** Start the EC2 instance and the standby node should start on dbprim00.

Test Case 3: Stop the Db2 Instance on the Primary Instance

**Procedure:** Log in to the Db2 primary instance (dbprim00) as db2<sid> (db2stj) and run db2stop force.

```
db2stj> db2stop force
02/12/2020 12:40:03 0 0 SQL1064N DBSTOP processing was successful.
```

**Expected result:** The Db2 primary node will failover on primary instance. The standby remains on the standby instance. There is a failed resource alert.
Test Case 4: End the Db2 Process (db2sysc) on the Node that Runs the Primary Database

Procedure: Log in to the Db2 primary instance as root and run `ps -ef|grep db2sysc`. Note the PID and then end it.

```
dbprim00:~ # ps -ef|grep db2sysc
  db2stj   11690 11688  0 19:27 ?        00:00:02 db2sysc 0
  root     15814  4907  0 19:31 pts/0    00:00:00 grep --color=auto db2sysc
[root@dbprim00 ~]# kill -9 11690
```

Expected result: The Db2 primary node is restarted on the primary instance. The standby node remains on the standby instance. There is a failed resource alert.

Followup action: Clear the failed cluster action.

```
dbsec00:~ # crm resource cleanup
Waiting for 1 reply from the CRMd. OK
```
**Test Case 5: End the Db2 Process (db2sysc) on the Node that Runs the Standby Database**

**Procedure:** Log in to the standby DB instance (dbsec00) as root, then run `ps -ef | grep db2sysc`. Note the PID and then end it.

```bash
dbsec00:~ # ps -ef | grep db2sysc
db2stj 16245 16243 0 19:23 ?        00:00:04 db2sysc 0
dbsec00:~ # kill -9 16245
```

**Expected result:** The db2sysc process is restarted on the standby DB instance. There is a monitoring failure event recorded in the cluster.

```bash
dbsec00:~ # crm status
Stack: corosync
Current DC: dbsec00 (version 1.1.19+20181105.ccd6b5b10-3.16.1-1.1.19+20181105.ccd6b5b10) - partition with quorum
Last updated: Sat Apr 25 19:40:23 2020
Last change: Sat Apr 25 19:23:04 2020 by root via crm_resource on dbprim00

2 nodes configured
4 resources configured

Online: [ dbprim00 dbsec00 ]

Full list of resources:

- `res.AWS_STONITH (stonith:external/ec2): Started dbprim00`
- `res.AWS_IP (ocf::suse:aws-vpc-move-ip): Started dbprim00`
- `Master/Slave Set: msl_db2_db2stj_STJ [rsc_db2_db2stj_STJ] Masters: [ dbprim00 ]`
- `Slaves: [ dbsec00 ]`

Followup action: Clear the failed cluster action.

```bash
dbsec00:~ # crm resource cleanup
Waiting for 1 reply from the CRMd. OK
```
Test Case 6: Simulating a Crash of the Node that Runs the Primary Db2

Procedure: Log in to the Db2 primary instance as root, then run `echo 'c' > /proc/sysrq-trigger`.

```
dbprim00:~ # echo 'c' > /proc/sysrq-trigger
Session stopped
   - Press <return> to exit tab
   - Press R to restart session
   - Press S to save terminal output to file
Network error: Software caused connection abort
```

Expected result: The primary Db2 should failover to standby node. The standby is in a stopped state on the previous primary (dbprim00).

```
[root@dbsec00 ~]# crm status
Cluster name: db2ha
Stack: corosync
Current DC: dbsec00 (version 1.1.18-11.el7_5.4-2b07d5c5a9) - partition with quorum
Last updated: Fri Feb 21 15:38:43 2020
Last change: Fri Feb 21 15:33:17 2020 by hacluster via crmd on dbsec00

2 nodes configured
4 resources configured

Online: [ dbprim00 dbsec00 ]

Full list of resources:

clusterfence   (stonith:fence_aws)    Started dbsec00
Master/Slave Set: Db2_HADR_STJ-master [Db2_HADR_STJ]
   Masters: [ dbsec00 ]
   Stopped: [ dbprim00 ]
db2-oip        (ocf::heartbeat:aws-vpc-move-ip)    Started dbsec00

Failed Actions:
   * Db2_HADR_STJ_start_0 on dbprim00 'unknown error' (1): call=15, status=complete,
     exitreason=''
     last-rc-change='Fri Feb 21 15:38:31 2020', queued=0ms, exec=7666ms
```
Daemon Status:
corosync: active/enabled
pacemaker: active/enabled
pcsd: active/enabled

Followup action: Start the EC2 instance and then start Db2 as standby on the standby instance as you did in Test Case 2 (p. 143).

FAQ

Question: Can I use Database Migration Service to migrate and deploy SAP NetWeaver on IBM Db2 based applications?

Answer: No, AWS DMS supports IBM Db2 as a source, but it is not certified by SAP for SAP NetWeaver-based applications.

Contributors

Contributors to this document include:

- Manas Srivastava Sr. Partner SA, SAP
- Somckit Khemmanivanh Sr. SysDev, SAP

Additional Reading

SAP on AWS technical documentation:

- SAP on AWS documentation
- SAP on AWS Whitepapers
- SAP on AWS Blog

SAP documentation:

- SAP Knowledge Base
- SAP Product Availability Matrix
- SAP Quick Sizer
- TCP/IP Ports of All SAP Products

Document Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>December 2020</td>
<td>First publication</td>
</tr>
</tbody>
</table>
Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

© 2020 Amazon Web Services, Inc. or its affiliates. All rights reserved.
Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

The software included with this document is licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with the License. A copy of the License is located at http://aws.amazon.com/apache2.0/ or in the "license" file accompanying this file. This code is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

© 2020 Amazon Web Services, Inc. or its affiliates. All rights reserved.