

SAP on Amazon Web Services Backup and Recovery Guide



Created by: Amazon Web Services
sap-on-aws@amazon.com
Version: 2.2

Document History

Before proceeding, make sure you have the latest version of this document. You can find the latest version on the AWS website:

<http://aws.amazon.com/sap>

The latest errata to this document can be found in [SAP Note 1677381](#).

Version	Date	Description
1.0	8/15/2012	Document created
2.0	9/18/2012	Sybase ASE Database additions
2.1	7/10/2013	Consolidated SAP notes, minor edits
2.2	12/29/2014	Updated content to reflect AWS platform enhancements

Contents

About this Guide	5
What Is Not Included in This Guide	5
Prerequisite Documentation	5
SAP on Amazon Web Services	5
General SAP backup and restore recommendations.....	5
SAP on SQL Server	6
SAP on DB2 LUW.....	6
SAP on MaxDB	6
SAP on Sybase ASE.....	6
SAP on Oracle DBMS.....	6
Overview of AWS Services for Backup and Recovery	7
Backup Destination on AWS	8
Backup and Recovery of Non-Production Systems	9
Create Image Backup Method	9
How to Execute an Ad-Hoc Backup Using Create Image	10
How to Schedule a Backup Using Create Image.....	10
How to Restore a System That Was Backed up Using Create Image.....	11
Snapshot Backups of Individual Primary Amazon EBS Volumes	11
Backup and Recovery of Production Systems	12
Database Backup to Secondary Amazon EBS Volumes	13
SAP File System Backup on Microsoft Windows	13
SAP File System Backup on Linux.....	14
Microsoft SQL Server on Microsoft Windows	14
IBM DB2 LUW on Linux or Windows.....	15
SAP MaxDB on Linux or Windows	16
SAP Sybase ASE on Linux or Microsoft Windows	17
Backup and Recovery Without Third-Party Backup Software	18
Copying Backup Data to, and Retrieving Backup Data from, Amazon S3	18
Creating a Snapshot of a Backup Amazon EBS Volume	19
Backup and Recovery Using Third-Party Backup Software.....	20

Using Third-Party Backup Software on AWS.....	20
Appendix 1: Linux System Backup Using TAR	21
Appendix 2: IBM DB2 Database Backup to a Local File System	23
DB2 Rollforward Recovery	23
To Enable Rollforward Recovery and Set Up DB2 Logfile Management.....	23
Example: DB2 LUW Full Online Data and Log Backup	24
Appendix 3: DB2 LUW Database Restore and Recovery	26
Example: DB2 Restore and Recovery from Amazon EBS Backup Volume Snapshots	26
Appendix 4: SAP MaxDB Database Backup to a Local File System	28
Example: Full Online Data and Log Backup for MaxDB	28
Create MaxDB Backup Templates	28
Option 1: Database and Log Backup Using MaxDB Database Studio	29
Option 2: Database and Log Backup Using DBMCLI	29
Schedule Hourly Automatic Log Backup Using Database Studio.....	30
Create a Snapshot to Send the Backup to Amazon S3.....	30
Appendix 5: SAP MaxDB Database Restore and Recovery	31
Example: MaxDB Restore and Recovery from Amazon EBS Backup Volume Snapshots	31
Option 1: Database Restore and Recovery Using MaxDB Database Studio.....	32
Option 2: Database Restore and Recovery Using MaxDB DBMCLI.....	33
Appendix 6: Sybase ASE Database Backup to a Local File System	36
Ensuring Recoverability of the Sybase ASE Database Server	36
Preparation	36
Automating Database and Transaction Log Dumps	37
Appendix 7: Sybase ASE Server Recovery	38

About this Guide

This guide provides an overview of how to back up SAP systems on Amazon Web Services (AWS). This guide is not intended to replace any of the standard SAP or RDBMS documentation. It focuses on the essential differences in backing up SAP systems on AWS as compared to traditional infrastructure.

Different backup options and methods are provided for non-production systems (such as sandbox, training, demo, proof-of-concept systems) and production systems.

What Is Not Included in This Guide

Apart from some examples, this guide does not include detailed instructions on how to execute database backups using either native RDBMS backup/recovery features or third-party backup tools. Please refer to the standard SAP and RDBMS documentation or the documentation provided by the backup software vendor.

Backup schedules, frequency, and retention periods are primarily based on your system type and business requirements. Please refer to the standard SAP documentation for guidance on these topics.

Prerequisite Documentation

This guide does not provide detailed instructions on how to execute SAP system and database backups or provide recommendations for backup frequencies or retention. For this information, please refer to the standard SAP and database vendor documentation listed in the following sections.

SAP on Amazon Web Services

This document assumes that you are already familiar with implementing and operating SAP solutions on the AWS infrastructure. It is highly recommended that you read *Implementing SAP Solutions on AWS* before continuing. This and other AWS guides for SAP can be found at <http://aws.amazon.com/sap/resources>.

Table 1 lists the available SAP notes for deploying SAP on the AWS infrastructure; the latest errata to this guide can be found in [SAP Note 1677381](#).

Table 1: SAP Notes for Deploying SAP on AWS

SAP Note #	Description
1588667	SAP on AWS: Overview of related SAP Notes and Web-Links
1656099	SAP on AWS: Supported SAP, DB/OS and AWS EC2 products
1656250	SAP on AWS: Support prerequisites
1656249	SAP on AWS: Initial implementation guidelines
1677381	SAP on AWS: Backup and Recovery guidelines

General SAP backup and restore recommendations

- SAP Help Portal: [Technical Operations Manual – Backup and Recovery](#)

SAP on SQL Server

- SAP Community Network: [SAP with SQL Server Best Practices Guide](#)
- SAP Help Portal: [Backup with MS SQL Server](#)
- Microsoft TechNet Library: [Backup and Recovery Overview for Windows Server 2008 R2](#)
- Microsoft MSDN Library: [Backup and Restore of SQL Server Databases](#)

SAP Note #	Description
44449	Backup strategies with the SQL Server
1297986	Backup and Restore strategy for MS SQL Server

SAP on DB2 LUW

- SAP Community Network: [Database Administration Guide SAP on IBM DB2 for Linux, UNIX, and Windows](#)
- SAP Community Network: [IBM DB2 Universal Database for UNIX and Windows - New Log File Management](#)
- SAP Community Network: [A Practical Guide to Backup and Recovery of IBM DB2 for Linux, UNIX and Windows in SAP Environments](#)

SAP on MaxDB

- SAP Community Network: [MaxDB Backup and Restore](#)
- SAP MaxDB Library: [Database Studio – Backing up Databases: Overview](#)

SAP Note #	Description
1377148	FAQ: SAP MaxDB Backup and Recovery
767598	Available SAP MaxDB documentation

SAP on Sybase ASE

- Sybase Infocenter: [System Administration Guide, Volume 2](#)
- SAP Community Network: [Getting Started with the Sybase Database and the SAP System](#)
- [SAP DBA Guide: Sybase Adaptive Server Enterprise](#)

SAP Note #	Description
1585981	SYB: Ensuring Recoverability for Sybase ASE
1611715	SYB: How to restore a Sybase ASE database server (Windows)
1618817	SYB: How to restore a Sybase ASE database server (UNIX)
1588316	SYB: Configure automatic database and log backups

SAP on Oracle DBMS

- SAP on Oracle - Backup and Recovery: <http://scn.sap.com/docs/DOC-7845>
- Oracle Recovery Manager (RMAN): <http://www.oracle.com/technetwork/database/features/availability/rman-overview-096633.html>


SAP Note #	Description
1430669	BR*Tools support for Oracle 11g
2087004	BR*Tools support for Oracle 12c

Overview of AWS Services for Backup and Recovery

This section provides an overview of the different AWS services that are required for the backup and recovery of SAP systems running on AWS. For a more detailed description of each service, please refer to the links provided.

Amazon Simple Storage Service (Amazon S3) – <http://aws.amazon.com/s3>

Amazon S3 provides a highly durable storage infrastructure designed for mission-critical and primary data storage. Amazon S3 is designed to provide 99.999999999% durability and 99.99% availability over a given year.

 Amazon S3 is in the center of any SAP backup and recovery solution on AWS. All backup methods described in this document rely on backup data being stored directly or indirectly in Amazon S3.

Amazon Elastic Compute Cloud (Amazon EC2) – <http://aws.amazon.com/ec2>

Amazon EC2 is an IT infrastructure platform that provides on-demand access to virtual Linux and Microsoft Windows servers, storage, networking, and other infrastructure services.

Components of Amazon EC2

Amazon Machine Image (AMI)

An AMI is the server template that is used to launch a new instance (virtual server). The AMI contains the base operating system (e.g., Linux or Microsoft Windows) on top of which you can install SAP software. A large selection of public AMIs is available from Amazon and the Amazon EC2 community, and it is possible to create your own AMIs.

Instance

After an AMI is launched, the resulting running system is called an *instance*.

Regions and Availability Zones

Amazon EC2 locations are composed of regions and Availability Zones.

Availability Zones are distinct locations that are engineered to be insulated from failures in other Availability Zones, and which provide inexpensive, low-latency network connectivity to other Availability Zones in the same region.

Regions consist of one or more Availability Zones, are geographically dispersed, and will be in separate geographic areas or countries. Amazon EC2 is currently available in 11 regions. For a list of regions and the products and services available in each, see [Products and Services by Region](#).

Amazon Elastic Block Store (Amazon EBS) – <http://aws.amazon.com/ebs>

Amazon EBS provides persistent block-level storage volumes for use with Amazon EC2 instances. Amazon EBS volumes offer off-instance storage that persists independently from the life of an instance. Amazon EBS provides highly available, highly reliable storage volumes that can be attached to an Amazon EC2 instance and exposed as a device within the instance's guest operating system.

Amazon EBS Snapshot - <http://aws.amazon.com/ebs/details>

Amazon EBS provides the ability to create point-in-time snapshots of volumes, which are persisted to Amazon S3. Snapshots can be used as the starting point for new Amazon EBS volumes as well as to protect data for long-term durability. If you make periodic snapshots of a volume, the snapshots are incremental so that only the blocks on the device that have changed since your last snapshot are saved in the new snapshot. Even though snapshots are saved incrementally, the snapshot deletion process is designed so that you need to retain only the most recent snapshot in order to restore the volume.

AWS Import/Export– <http://aws.amazon.com/importexport>

AWS Import/Export accelerates moving large amounts of data into and out of AWS using portable storage devices for transport. AWS transfers your data directly onto and off storage devices using Amazon's high-speed internal network and bypassing the Internet. For significant data sets, AWS Import/Export is often faster than Internet transfer and more cost effective than upgrading your connectivity.

AWS Direct Connect – <https://aws.amazon.com/directconnect/>

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 datacenter, office, or colocation environment. In many cases, this can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than Internet-based connections.

Amazon Glacier – <http://aws.amazon.com/glacier>

Amazon Glacier is an extremely low-cost storage service that provides secure and durable storage for data archiving and backup. In order to keep costs low, Amazon Glacier is optimized for data that is infrequently accessed and for which retrieval times of several hours are suitable. With Amazon Glacier, customers can reliably store large or small amounts of data for as little as \$0.01 per gigabyte per month—a significant savings compared to on-premises solutions.

Backup Destination on AWS

The primary difference between backing up SAP systems on AWS compared to a traditional on-premises infrastructure is the backup destination. The typical backup destination used with on-premises infrastructure is tape. On AWS, backups are stored in Amazon S3 instead of on tape. There are many benefits to storing backups in Amazon S3 vs. tape. Backups stored in Amazon S3 are automatically stored offsite from the source system since data in Amazon S3 is replicated across multiple facilities within the AWS region.

There are primarily two different methods to store backups in Amazon S3. The first method is to back up data directly into Amazon S3. The second method involves backing up your data to a locally attached Amazon EBS volume, and then subsequently copying this data into Amazon S3. Details on how to copy backup data from an Amazon EBS volume into Amazon S3 are provided later in this document.

Backup and Recovery of Non-Production Systems

This section provides backup options for non-production systems. Examples of non-production systems are:

- Demo systems
- Training systems
- Sandbox systems
- Proof-of-concept systems
- Trial systems

Systems of this type share the following backup requirements:

- Infrequent backups (1-2 times per week)
- Point-in-time recovery isn't required
- Downtime to complete a backup can be tolerated
- Simple low cost backup solutions

Create Image Backup Method

The Amazon EC2 Create Image function can be used to create full (operating system, application, and database file systems) offline backups of a complete SAP system. Executing the Create Image function creates and registers a new Amazon Machine Image (AMI) of the running instance and creates snapshots of all Amazon EBS volumes attached to the instance. Create Image can also be used to copy complete SAP systems with very little effort. The AMI created during the Create Image process can then be used to launch additional Amazon EC2 instances that are identical to the source SAP system.

Please note that in order to provide a consistent backup, the Amazon EC2 instance is powered down at the beginning of the Create Image process to confirm that all I/O to the Amazon EBS volumes is quiesced. The Amazon EC2 instance is stopped only for a short period of time and is restarted while Amazon EBS snapshot synchronization is continued in the background.

Figure 1 illustrates the Create Image process.

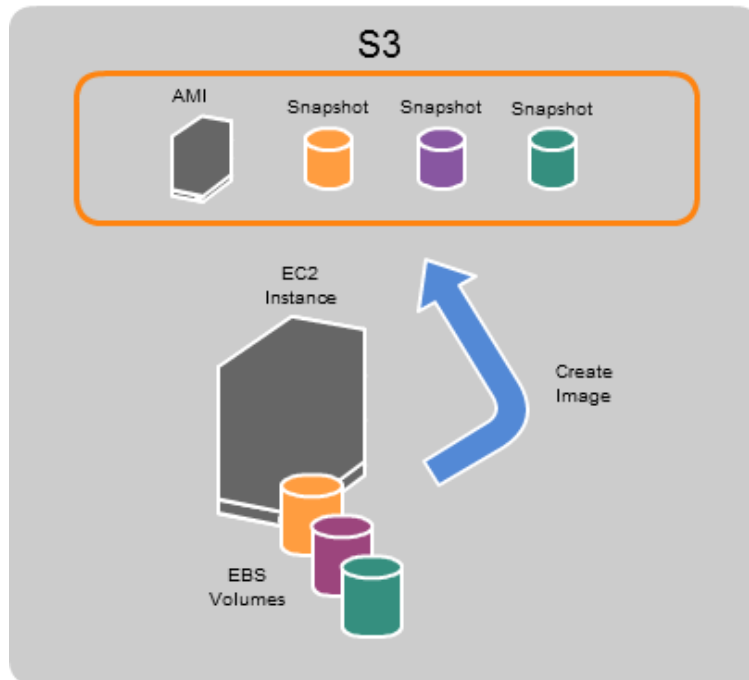


Figure 1: The Create Image (AMI) Process

How to Execute an Ad-Hoc Backup Using Create Image

You can execute the Create Image function from the AWS Management Console by right-clicking the instance you would like to back up, and then choosing **Image, Create Image**.

For detailed documentation for the Create Image process, please read [Creating an Amazon EBS-Backed AMI](#) in the *Amazon EC2 User Guide*.

! Creating an Amazon EC2 image will shut down and restart the Amazon EC2 instance (VM and operating system) to create consistent snapshots. You need to make sure that SAP applications and databases are cleanly shut down before using this command. After the Amazon EC2 instance restarts, databases and SAP applications may need to be restarted to resume SAP operation.

How to Schedule a Backup Using Create Image

You can schedule backups using the AWS command line interface (AWS CLI). The AWS CLI allows you to manage Amazon EC2 resources from a Windows, OS X, Linux, or Unix command shell. The CLI includes the `aws ec2 create-image` command, which is used to create an AMI from an existing Amazon EC2 instance.

Here is the syntax and an example of how to create a new AMI using the `aws ec2 create-image` command:

Syntax:

```
aws ec2 create-image --name <value> --instance-id <value>
```

Example:

```
<stopsap on EC2 instance i-eo9483uj>  
  
aws ec2 create-image --name "ECC_Demo_20120401 --instance "i-  
eo9483uj"  
  
<startsap on EC2 instance i-eo9483uj>
```

How to schedule the Create Image process:

1. Install and configure the AWS CLI on a server on your corporate network or on another Amazon EC2 instance. (This can be a dedicated administration system used for tasks like this.)
2. Create a shell or batch script that contains the `aws ec2 create-image` command with the necessary parameters to create an image of the instance you wish to back up.
3. Schedule the batch script to run when you would like the Create Image process to take place, using **cron** on Linux or the Task Scheduler built into Microsoft Windows.

For detailed information on the `aws ec2 create-image` command and the AWS CLI, including download information, see the following pages in the AWS documentation:

- [create-image reference page](#) in the *Amazon AWS CLI Command Reference*
- [Getting Started with the CLI](#)

How to Restore a System That Was Backed up Using Create Image

Restoring a system that was backed up using the Create Image method is very simple. Since the Create Image process creates a complete image (AMI) of the instance being backed up by creating snapshots of all attached Amazon EBS volumes, all that is needed to restore a system is to launch a new instance from the latest AMI version.

For detailed instructions on how to launch an instance from an AMI, see [Launch Your Instance](#) in the *Amazon EC2 User Guide*.

After the complete system has been restored from an AMI, the SAP database(s) can be rolled forward as described in the next section.

Snapshot Backups of Individual Primary Amazon EBS Volumes

Instead of creating a complete AMI system backup as described in the previous section, snapshots of individual Amazon EBS volumes can be leveraged to create more fine-grained snapshot backups.

The Create Snapshot process of individual Amazon EBS volumes is illustrated in Figure 2.

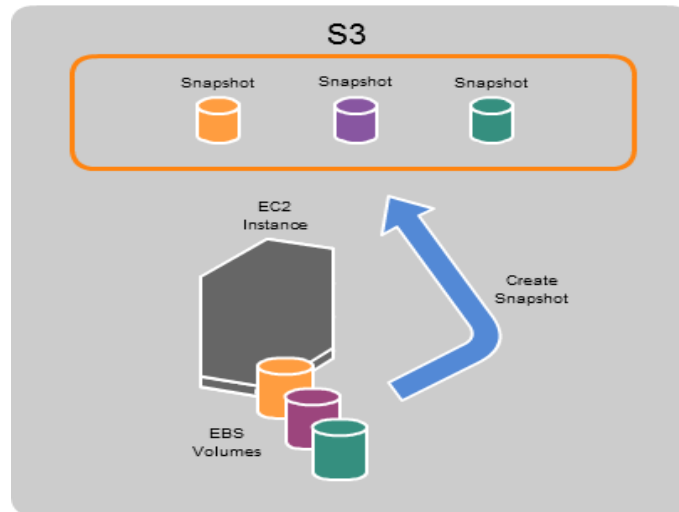


Figure 2: Snapshots of Individual Primary Amazon EBS Volumes

You can create a consistent snapshot of individual Amazon EBS volumes on a running system as long as you can make sure that no I/O is taking place on the Amazon EBS volumes, and that the (file system) cache is flushed properly to the underlying Amazon EBS volumes. The easiest and safest way to assure Amazon EBS volume consistency is to stop processes, such as the SAP application and its database, which access the Amazon EBS volumes, and to dismount their file systems properly before creating snapshots of underlying volumes.

Alternatively, combinations of DBMS and file system techniques can be used to temporarily freeze file system I/O for a short period of time to create snapshots of the underlying Amazon EBS volumes, and to resume I/O after consistent snapshot creation has finalized.

⚠ Direct primary Amazon EBS volume snapshot techniques of SAP systems and their databases are complex to set up correctly. Therefore, a discussion of those techniques is not within the scope of these general backup and restore guidelines.

Backup and Recovery of Production Systems

The backup options covered in this section address the following backup requirements that are common for production systems:

- Frequent backups based on a schedule
- Online database data backup
- Database transaction log backup
- Point-in-time database recovery
- Closely synchronized file system and database backups

For production systems, we recommend a regular restoration of backups to a separate system, so that:

- Restore and recovery procedures, and Service Level Agreement (SLA) objectives can be validated.
- Restored data can be checked for consistency using the DBMS tools listed in Table 2.

Table 2: Overview of DBMS Tools for Checking Data Consistency and Integrity

DBMS	Tool
Microsoft SQL Server	DBCC CHECKDB
IBM DB2	DB2DART
SAP MaxDB	CHECK DATA
Sybase ASE	DBCC CHECKDB

Database Backup to Secondary Amazon EBS Volumes

The backup options covered in this section involve first writing the backup data to a local Amazon EBS backup volume, and then copying the backup data to Amazon S3 so that the data is protected in case of failure of the Amazon EBS backup volume.

Figure 3 shows how backups are first created on a local Amazon EBS volume, and then copied to Amazon S3.

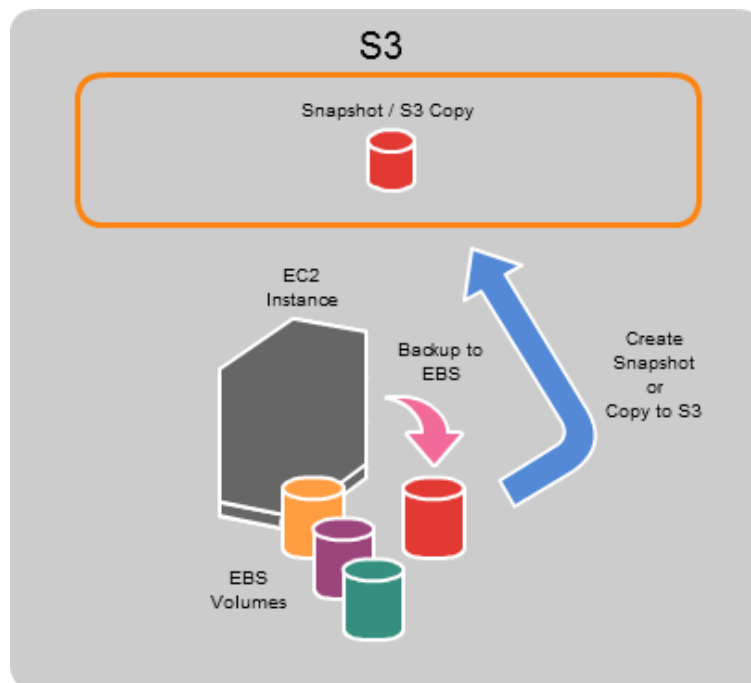


Figure 3: Backup to Amazon EBS, and Snapshot/Copy to Amazon S3

SAP File System Backup on Microsoft Windows

Backup and recovery of the operating system and critical SAP file systems is accomplished by using the Windows Server Backup feature built into Microsoft Windows. This feature is not installed by default on the standard Windows Server AMI, but can be easily added by following these steps:

On Windows Server 2008 R2:

1. Choose the **Start** button, and then right-click **Computer**.
2. Choose **Manage**.
3. On the left, choose **Server Manager**.
4. On the right, scroll down to **Features Summary**, and then choose **Add Features**.
5. Choose **Windows Server Backup Features**, and then choose **Install**.

On Windows Server 2012:

1. Open **Server Manager**.
2. Choose **Manage**, add roles and features.
3. Choose role-based or feature-based installation.
4. Choose your server.
5. Choose **Next** to navigate past role selection.
6. On the features summary, scroll down and select **Windows Server backup**.
7. Click **Install**.

Once the Windows Server Backup Feature is installed, you can schedule backups of drives and directories by using the Windows Server Backup wizard.

While scheduling a backup in the Windows Server Backup wizard, you will be asked to choose a **Destination Type**. Choose **Local Drives**, and then select the drive that is the Amazon EBS backup volume (Windows drive) and the folder you wish to back your data up to.

SAP File System Backup on Linux

Open source tools for backing up data on Linux include `tar`, `cpio`, and `rsync`; a detailed example is given in [Appendix 1: Linux System Backup Using TAR](#).

For instructions on how to create a full SAP system backup on Linux with TAR, please see [Backing Up and Restoring your SAP System on UNIX](#) in the SAP Library (use the file system example herein, with a file location on an Amazon EBS backup volume).

Microsoft SQL Server on Microsoft Windows

Table 3 shows a sample database layout for SQL Server, where each Windows drive is backed by a single Amazon EBS volume. For additional information on recommended Amazon EBS/drive configurations for SAP systems on AWS, please refer to *Implementing SAP Solutions on AWS* referenced previously in the [Prerequisite Documentation](#) section.

Drives X:\, Y:\, and Z:\ are the secondary drives that will hold the file system, database data, and database log backups to disc. After a backup to one of these drives has finished, it can be sent to Amazon S3 by creating a snapshot of the underlying Amazon EBS volume, or by copying the individual backup files directly to Amazon S3, as described in the section [Backup and Recovery Without Third-Party Backup Software](#).

Table 3: Sample SQL Server Drive Layout with Backup Amazon EBS Volumes

Windows Drive	Use
C:\	Root volume (included with Amazon EC2 instance)
D:\	SAP / Microsoft SQL Server software
S:\	Swap
E:\	Database log file
F:\ G:\ H:\ I:\	Database data files 1-4
X:\	File system backups
Y:\	Database log backups
Z:\	Database data backups

The backup and recovery of the SAP database and transaction log are accomplished using the native SQL Server backup functionality that can be scheduled and executed either interactively from the SQL Server Management Studio or via a script using Transact-SQL commands.

As with the file system backup described earlier, the only special consideration when scheduling and executing SQL Server backups on AWS is the backup destination. Instead of choosing the backup destination **Tape**, select **Disk** and point to the drive for the Amazon EBS backup volume that you designated (in our example, Y:\ for log and Z:\ for data backups).

For database restore and recovery:

1. First check if all required data and transaction log backups are still available on the designated Amazon EBS backup volumes (Y:\ and Z:\ in our example).
2. If the available backups are not usable or are incomplete, retrieve the missing data and transaction log backups from Amazon S3 as described in the section [Copying Backup Data to, and Retrieving Backup Data From, Amazon S3](#).
3. Perform SQL Server Restore and Recovery, using the data and transaction log backups that are available on the Amazon EBS volumes

IBM DB2 LUW on Linux or Windows

Table 4 shows an example database layout for IBM DB2. For additional information on recommended Amazon EBS/drive configurations for SAP systems on AWS, please refer to *Implementing SAP Solutions on AWS* referenced previously in the [Prerequisite Documentation](#) section.

The secondary file systems `/os_exe_backups` (Windows drive X:\) and `/db_data_backups` (Windows drive Z:\) will hold system and database data backups to disk. The file system `/db2/<SID>/log_archive` (Windows drive Y:\) is used to let DB2 automatically archive its database transaction logs, which is a process managed by the DB2 RDBMS itself.

After a backup to `/os_exe_backups` or `/db_data_backups` has finished, or a transaction log has been archived to `/db2/<SID>/log_archive` by the DB2 RDBMS, it can be sent to Amazon S3 either by creating a snapshot of the underlying Amazon EBS volume, or by copying the individual files directly to Amazon S3, as described in the section [Backup and Recovery Without Third-Party Backup Software](#).

Table 4: Sample DB2 File System Layout with Backup Amazon EBS Volumes on Linux or Windows

Linux file system	Windows Drive(s)	Use
/	C:\	Root volume (included with Amazon EC2 instance)
/sapmnt /usr/sap /db2	D:\	SAP and DB2 software
<swap>	S:\	Swap
/db2/<SID>/log_dir	E:\	Active database transaction logs
/db2/<SID>/sapdata1-4	F:\ G:\ H:\ I:\	Database data file systems 1-4
/os_exe_backups	X:\	File system backups
/db2/<SID>/log_archive	Y:\	Archived database transaction logs
/db2_data_backups	Z:\	Database data backups

An example of how to create DB2 database backups to /db_data_backups is given in [Appendix 2: IBM DB2 Database Backup](#).

For detailed instructions on DB2 database and transaction log backups, and DB2 restore and recovery in SAP environments, please refer to [Database Administration Guide -- SAP on IBM DB2 for Linux, UNIX, and Windows](#) on the SAP Community Network website.

SAP MaxDB on Linux or Windows

Table 4 shows an example database layout for SAP MaxDB. For additional information on recommended Amazon EBS/drive configurations for SAP systems on AWS, please refer to *Implementing SAP Solutions on AWS* referenced previously in [Prerequisite Documentation](#).

The secondary file systems /os_exe_backups, /db_log_backups, and /db_data_backups (Windows drives X:\, Y:\ and Z:\) will hold system, database logs, and database data backups to disk.

After a backup to one of these file systems or Windows drives has finished, it can be sent to Amazon S3 either by creating a snapshot of the underlying Amazon EBS volume, or by copying the individual files directly to Amazon S3, as described in the section [Backup and Recovery Without Third-Party Backup Software](#).

Table 5: Sample MaxDB File System Layout with Amazon EBS Backup Volumes on Linux or Windows

Linux file system	Windows Drive	Use
/	C:\	Root volume (included with Amazon EC2 instance)
/sapmnt /usr/sap /sapdb	D:\	SAP and MaxDB software
<swap>	S:\	Swap
/sapdb/<SID>/saplog	E:\	Database transaction log
/sapdb/<SID>/sapdata1-4	F:\ G:\ H:\ I:\	Database data file system 1-4

Linux file system	Windows Drive	Use
/os_exe_backups	X:\	File system backups
/db_log_backups	Y:\	Database log backups
/db_data_backups	Z:\	Database data backups

An example of how to create MaxDB database backups to /db_data_backups and MaxDB log backups to /db_log_backups is given in [Appendix 4: SAP MaxDB Database Backup to a Local File System](#).

An example of how to restore and recover a MaxDB database from Amazon EBS backup volume snapshots is given in [Appendix 5: SAP MaxDB Database Restore and Recovery](#).

SAP Sybase ASE on Linux or Microsoft Windows

Table 4 shows an example database layout for SAP Sybase ASE. For additional information on recommended Amazon EBS/drive configuration for SAP systems on AWS, please refer to *Implementing SAP Solutions on AWS* referenced previously in [Prerequisite Documentation](#).

The secondary file systems /os_exe_backups, /db_log_backups, and /db_data_backups (Windows drives X:\, Y:\ and Z:\) will hold system, database log, and database data backups to disk.

After a backup to one of these file systems or Windows drives has finished, it can be sent to Amazon S3 either by creating a snapshot of the underlying Amazon EBS volume, or by copying the individual files directly to Amazon S3, as described in the section [Backup and Recovery Without Third-Party Backup Software](#).

Table 6: Sample Sybase ASE File System Layout with Amazon EBS Backup Volumes on Linux or Windows

Linux file system	Windows Drive	Use
/	C:\	Root volume (included with Amazon EC2 instance)
/sapmnt /usr/sap /sybase	D:\	SAP and Sybase ASE software
<swap>	S:\	Swap
/sapdb/<SID>/saplog_1	E:\	DB Transaction Log
/sapdb/<SID>/sapdata_1-N	F:\ G:\ H:\ I:\	DB Data File System 1-4
/os_exe_backups	X:\	File System backups
/db_log_backups	Y:\	DB Log backups
/db_data_backups	Z:\	DB Data backups

Backup and recovery of the Sybase ASE database and transaction log are accomplished using the native Sybase ASE backup functionality, which can be executed interactively or via a script using Transact-SQL commands.

As with the file system backup described earlier, the only special consideration when scheduling and executing Sybase ASE backups on AWS is choosing the correct backup destination. You need to save the Sybase ASE dumps to the file system or drive that is on the Amazon EBS backup volume that you have


designated for backups—in our example, `/db_log_backups` (drive Y:\) for log- and `/db_data_backups` (drive Z:\) for data backups..

The `/db_data_backups` file system (drive Z:\) will hold the dump images of the following databases:

- `<DBSID>` database (SAP database)
- `master` database
- `sybssystemprocs` database
- `sybmgmt` database
- `saptools` database

You can use the `DUMP DATABASE` command to create these database dump images.

The `/db_log_backups` file system (drive Z:\) will hold dump images of the transaction log. You can use the `DUMP TRANSACTION` command to create these transaction log dump images.

 Make sure that a complete sequence of transaction log archives is available at all times! SAP Notes [1585981](#) and [1588316](#) provide details on how this can be achieved and automated.

For database restore and recovery:

1. First check if all required data and transaction log dumps are still available on the designated Amazon EBS backup volumes—in our example, `/db_data_backups` (drive Z:\) and `/db_log_backups` (drive Y:\).
2. If the available backups are not usable or are incomplete, retrieve the missing data and transaction log dumps from Amazon S3 as described in the section [Copying Backup Data to, and Retrieving Backup Data From, Amazon S3](#).
3. Perform Sybase ASE restore and recovery, using the data and transaction log dumps that are available on the Amazon EBS volumes.

For detailed instructions on how to create Sybase ASE database and transaction log dump images, and perform restore and recovery by using the Sybase ASE Transact-SQL commands, please refer to the [SAP DBA Guide: Sybase Adaptive Server Enterprise](#) and the SAP notes referenced in [Prerequisite Documentation](#).

Backup and Recovery Without Third-Party Backup Software

Once you have created a backup on an Amazon EBS volume, you need to copy the backup data to Amazon S3 to protect the data against Amazon EBS volume failure.

Copying Backup Data to, and Retrieving Backup Data from, Amazon S3

Amazon S3 stores data as objects within buckets. An object comprises a file and optimally any metadata that describes that file. We recommend that you create a dedicated Amazon S3 bucket to store your backups. You can use one bucket to store backups from multiple systems. To help organize backups from different systems and different backup types (file system, database, transaction log, etc.) you can create multiple folders within a bucket.

For detailed instructions on how to create an Amazon S3 bucket, see [Create a Bucket](#) in the *Amazon S3 Getting Started Guide*.

AWS provides four methods to copy data directly to Amazon S3:

- AWS Management Console
- AWS CLI
- AWS SDK API
- Amazon S3 REST API

You do not have to develop your own command line interface to copy data to Amazon S3. An extensive AWS command line interface (CLI) with Amazon S3 support is available that you can incorporate into a script to copy data to, or retrieve data from, Amazon S3. For detailed information, see [Using Amazon S3 with the AWS Command Line Interface](#) in the AWS documentation.

The following examples of using the AWS CLI `aws s3` command are called from either a Linux shell or the Microsoft Windows command prompt.

To copy a backup stored on an Amazon EBS volume to Amazon S3:

```
aws s3 cp backup_file s3://my_bucket/my_folder/backup_file
```

To retrieve a backup from Amazon S3 to an Amazon EBS volume to be used for a restore:

```
aws s3 cp s3://my_bucket/my_folder/backup_file backup_file
```

Creating a Snapshot of a Backup Amazon EBS Volume

Instead of copying and retrieving single files to and from Amazon S3, you can create a snapshot of the complete Amazon EBS volume. The snapshot will automatically be stored in Amazon S3 and can subsequently be used to create new Amazon EBS volumes that contain the same data as stored in the snapshot.

The following sample commands, using the AWS CLI, are called from either a Linux shell or the Microsoft Windows command prompt.

To create a snapshot of an Amazon EBS volume that is automatically stored in Amazon S3:

```
aws ec2 create-snapshot --volume-id <value> --description <value>
```

To create a new Amazon EBS volume from an Amazon EBS snapshot stored in Amazon S3:

```
aws ec2 create-volume --snapshot-id <value> --availability-zone  
<value>
```

To attach an Amazon EBS volume to an Amazon EC2 instance:

```
aws ec2 attach-volume --volume-id <value> --instance-id <value> --  
device <value>
```

Backup and Recovery Using Third-Party Backup Software

This section discusses using third-party solutions to back up SAP systems running on AWS. Many of the same reasons for choosing third-party enterprise backup solutions for managing the backup and recovery of SAP systems running on traditional on-premises infrastructure also apply to backing up SAP systems on AWS:

- Centralized backup schedule management
- Automated data retention and disposition policy
- Clients and agents for various operating systems and databases
- Reporting

Using Third-Party Backup Software on AWS

Using an enterprise backup solution on AWS is similar to using it in a traditional on-premises infrastructure. The only special consideration, as discussed previously in this guide, is the backup destination. Many enterprise backup solutions support backup directly to Amazon S3, and many more support the use of a cloud gateway for virtualizing storage. For enterprise backup solutions that do not support backup directly to Amazon S3, you can first back up to a local Amazon EBS volume attached to the instance, and then copy the backup data to Amazon S3. Most enterprise backup solutions allow you to run scripts before and after the backup is run. One of the post-scripts you would run would be a script to copy the backup data to Amazon S3 as described previously in this section.

Figure 4 shows a direct backup to Amazon S3.

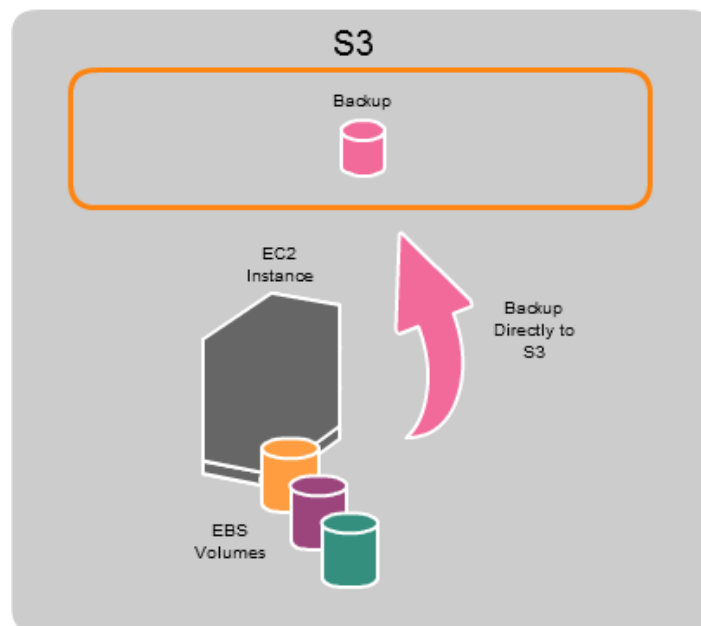


Figure 4: Direct Backup to Amazon S3

Appendix 1: Linux System Backup Using TAR

This appendix provides an example for creating a Linux file system backup that can be used in case of a full system loss.

NOTE: THIS OPERATING SYSTEM BACKUP CANNOT BE USED TO RESTORE THE DATABASE, AS DATABASE DATA AND TRANSACTION LOGS ARE SPECIFICALLY EXCLUDED FROM THE BACKUP. AFTER RESTORING THIS BACKUP, RESTORE AND RECOVERY OF THE DATABASE SHOULD FOLLOW.

The example procedure is as follows:

1. Make sure that you have enough space on the backup Amazon EBS volume file system
/os_exe_backups.

NOTE: DATABASE DATA, TRANSACTION LOGS AND MOUNTED BACKUP FILE SYSTEMS WILL BE EXCLUDED FROM THIS BACKUP.

2. Start the backup of operating system, SAP, and DBMS binaries and profiles.
 - a. Log on to the operating system, and start a TAR backup on the /os_exe_backups file system.

NOTE: THE USE OF COMPRESSION AFFECTS THE CPU UTILIZATION AND BACKUP TIME. USE THE --GZIP PARAMETER OF THE TAR COMMAND TO SPECIFY WHETHER YOU WANT TO USE COMPRESSION.

For database type of MaxDB, use this script as an example:

```
export exclusion_file=/os_exe_backups/backup-exclude-dirs.txt
export backup_file=/os_exe_backups/backup.tar.gz
export log_file=/os_exe_backups/backup.stdout
export error_log_file=/os_exe_backups/backup.stderr

#Create the exclusion file
#Exclude OS directories
>$exclusion_file
echo "/tmp" >>$exclusion_file
echo "/proc" >>$exclusion_file
echo "/sys" >>$exclusion_file
echo "/dev" >>$exclusion_file
#Exclude database files
echo "/sapdb/<SID>/sapdata1" >>$exclusion_file
echo "/sapdb/<SID>/saplog" >>$exclusion_file
#Exclude the backup directories
echo "/os_exe_backups" >>$exclusion_file
echo "/db_data_backups" >>$exclusion_file
echo "/db_log_backups" >>$exclusion_file

#Run the backup
cd /os_exe_backups
tar -v --gzip -cf $backup_file / --exclude-from=$exclusion_file >
$log_file 2> $error_log_file

#to monitor
#tail -f $log_file
#tail -f $error_log_file
```

NOTE: THE COMMAND HAS BEEN PROVIDED AS AN EXAMPLE; PLEASE TEST AND CHANGE AS NECESSARY.

For database type of DB2, use this script as an example:

```
export exclusion_file=/os-exe-backups/exclude-dirs.txt
export backup_file=/os-exe-backups/backup.tar.gz
export log_file=/os-exe-backups/backup.stdout
export error_log_file=/os-exe-backups/backup.stderr

#Create the exclusion file
#Exclude OS directories
>$exclusion_file
echo "/tmp" >>$exclusion_file
echo "/proc" >>$exclusion_file
echo "/sys" >>$exclusion_file
echo "/dev" >>$exclusion_file

#Exclude database-related filesystems
echo "/db2/<SID>/sapdata1" >>$exclusion_file
:
echo "/db2/<SID>/sapdata<n>" >>$exclusion_file
echo "/db2/<SID>/log_dir" >>$exclusion_file
echo "/db2/<SID>/log_archive" >>$exclusion_file

#Exclude the backup directories
echo "/os-exe-backups" >>$exclusion_file
echo "/db2-data-backups" >>$exclusion_file

#Run the backup
cd /os-exe-backups
tar -v --gzip -cf $backup_file / --exclude-
from=$exclusion_file > $log_file 2> $error_log_file

#to monitor
#tail -f $log_file
#tail -f $error_log_file
```

NOTE: THE COMMANDS HAVE BEEN PROVIDED AS AN EXAMPLE, PLEASE TEST AND CHANGE AS NECESSARY.

To send the backup to Amazon S3, create an Amazon EBS snapshot of the Amazon EBS backup volume that holds the /os_exe_backups file system.

We recommend that you tag the snapshot with a description like this:

```
<EC2-INSTANCE-ID>_OS_EXE_BACKUPS_<YYYY-MM-DD>
```

where <EC2-INSTANCE-ID> can be retrieved from the Amazon EC2 metadata web service at <http://169.254.169.254/latest/meta-data/instance-id>.

You can use Linux tools such as **curl** or **wget** to issue the above HTTP command from a local shell on the Amazon EC2 instance.

For more information on using Amazon EC2 instance metadata, see [Instance Metadata and User Data](#) in the *Amazon EC2 User Guide*.

Appendix 2: IBM DB2 Database Backup to a Local File System

DB2 Rollforward Recovery

Make sure that you have enabled your DB2 database for rollforward recovery. Rollforward recovery mode enables you to recover from a database backup to the most recent point in time by using archived database log files and is a prerequisite for taking DB2 online backups. If you have not configured your DB2 database for rollforward recovery mode, you cannot take online backups. Instead, you will need to shut down the SAP system and DB2 to take offline backups. For **production systems**, your database **must** be in rollforward recovery mode. For details, refer to the [Database Administration Guide SAP on IBM DB2 for Linux, UNIX, and Windows](#) on the SAP Community Network website.

The file system names listed in Table 4 will be used for the following examples.

To Enable Rollforward Recovery and Set Up DB2 Logfile Management

1. Stop SAP and DB2.
2. Enable rollforward recovery by updating the database configuration parameter LOGARCHMETH1:

```
su - db2<sid>
db2 update db cfg for <SID> using logarchmeth1
DISK:/db2/<SID>/log_archive
```

The database will now be placed in backup pending state. A full database backup must be taken.

From this point onwards DB2 will automatically archive log files from the /db2/<SID>/log_dir file system to the /db2/<SID>/log_archive file system.

3. Take a full database offline backup.

```
db2 backup database <sid> to "/db_data_backups" compress
```

4. Configure a retention period for your database backups and the corresponding database log files. For example, if you want to keep at least four database backups, and you want to remove surplus backups older than 30 days and all corresponding database log files, configure DB2 in the following way:

```
db2 update db cfg for <sid> using NUM_DB_BACKUP 4
db2 update db cfg for <sid> using REC_HIS_RETENTN 30
db2 update db cfg for <sid> using AUTO DEL REC OBJ ON
```

5. Start SAP.

Example: DB2 LUW Full Online Data and Log Backup

Make sure that the `/db_data_backups` file system (from Table 4) has enough storage space and proper read/write permissions.

Please remember to use the references mentioned in the section [Prerequisite Documentation](#) as primary documentation. In addition, you can consult the [SAP on DB2 LUW for Unix and Windows \(DB6\)](#) forum on the SAP Community Network for additional questions. The following sections merely provide examples, and should not be used as a general reference.

In this appendix, we'll discuss two options for backing up the DB2 LUW database:

- Option 1: Back up the database directly from the SAP system
- Option 2: Back up the database using the CLI

Option 1: Back up the database directly from the SAP system

1. Log in to the SAP system with an administrator credentials.
2. Run the transaction `/nDBACOCKPIT`.
3. On the left screen panel, navigate to **Jobs/DBA Planning Calendar**.
4. In the calendar area, select any cell representing time older than the current time, and click **Add**.
5. On the pop-up window, select **Database Backup to Device**, choose **Online** backup mode with the **Include Logs** option, and then enter `/db2_data_backups` in the **Device/Directory** field.
6. Click **Execute Immediately**.

Option 2: Back up the database using the CLI

1. Log in to the Amazon EC2 instance.
2. Run the following commands:

```
su - db2<sid>
#start online compressed backup including logs
db2 backup database <sid> online to "/db2_data_backups" compress \
include logs
```

3. Wait until the backup has completed, and you get the following message:

```
Backup successful. The timestamp for this
backup image is : 20111220221428
```

This message is provided in the following log file, located in the backup directory:

```
<SID>.0.db2<sid>.NODE0000.CATN0000.<datetime stamp>.001
```

4. Back up the database manager configuration required to rebuild the database:

```
su - db2<sid>
cd /db2_data_backups
db2cfexp <SID> cfg_backup.txt BACKUP
```


5. Back up the DB2 recovery history file:

```
su - db2<sid>
cp /db2/<SID>/db2<sid>/NODE0000/SQL00001/db2rhist.asc \
/db2_data_backups
```

DB2 transaction log file management

We recommend that you configure the DB2 transaction log management as mentioned in the previous example and described in [IBM DB2 Universal Database for UNIX and Windows - New Log File Management](#), section 2.2.1, on the SAP Community Network website.

As from DB2 LUW V9.5, automatic log file retention management can be configured in addition. This is described in the previous example and in the section “DB2 V9.5 and Higher Only: Automatic Log File and Backup Retention” in [Database Administration Guide -- SAP on IBM DB2 for Linux, UNIX, and Windows](#), on the SAP Community Network website.

The content of the `/db2/<SID>/log_archive` file system should be sent to Amazon S3 on a regular basis by creating a direct snapshot of the underlying Amazon EBS volume, or by copying added files individually to Amazon S3, optimally each time after a transaction log was written into the DB2-LOG-ARCHIVE file system. The snapshot of the Amazon EBS volume can be taken directly without dismounting the file system if this file system is the only one residing on a single Amazon EBS volume (no striping, etc.).

Create a snapshot to send the backup to Amazon S3

We recommend that you tag snapshots with a description; for example:

- `<SID>_DB2_DATA_BACKUPS_<YYYY-MM-DD>` for the database data backups.
- `<SID>_DB2_LOG_ARCHIVE_<YYYY-MM-DD-HH-MM-SS>` for the database archive log backups.

Appendix 3: DB2 LUW Database Restore and Recovery

The restoration and recovery of databases require careful planning and preparation. Take time for root-cause analysis so you can identify the most efficient recovery strategy before executing.

For example, you need to restore data from Amazon S3 only if recovery is not possible any longer from data and backups that are already available on the system. If a database needs to be recovered to the latest possible point in time, always make sure not to overwrite and destroy the latest database transaction logs. These are typically not yet archived or backed up to Amazon S3, so the latest transactions contained in these logs could get lost forever!

This guide does not intend to replace the original backup and restore documentation provided by database vendors and listed in [Prerequisite Documentation](#). Restoration and recovery scenarios are diverse, and they depend on the environment and the cause of failure, so we strongly recommend that you follow the original documentation in case of a real failure.

Example: DB2 Restore and Recovery from Amazon EBS Backup Volume Snapshots

1. If this is a disaster recovery, make sure that you have restored the operating system.
2. Mount the correct database backup on the local Amazon EC2 instance.

NOTE: THIS STEP IS REQUIRED ONLY IF THE ALREADY MOUNTED DATABASE BACKUP IS NOT ADEQUATE FOR RECOVERY, OR IF THE LATEST DATABASE BACKUP IS NO LONGER AVAILABLE ON THE AMAZON EC2 INSTANCE.

- a. Create an Amazon EBS volume based on an Amazon EBS snapshot:
 - I. On **Volumes**, click **Create Volumes**.
 - II. Specify the **Size** of the volume.
 - III. Select the same Availability Zone as the instance.
 - IV. On **Snapshot**, select the latest database backup.
 - V. Choose **Yes, Create**.
- b. Attach the volume to the Amazon EC2 instance.
- c. Mount the file system:
 - I. Log on to the Amazon EC2 instance using PuTTY or any other SSH client.
 - II. Mount the backup file system; for example:

```
vgscan
vgimport vgbbackup
vgchange vgbbackup -a y
mkdir /db_data_backups
mount /dev/vgbbackup/backups /db_data_backups
```

3. Mount the required database logs on the instance.

NOTE: THIS STEP IS REQUIRED ONLY IF THE AVAILABLE ARCHIVED LOGS ARE NOT SUFFICIENT FOR RECOVERY, OR IF THEY ARE NO LONGER AVAILABLE.

- a. Create an Amazon EBS volume based on an Amazon S3 log backup snapshot:
 - I. On **Volumes**, click **Create Volumes**.
 - II. Type the **Size** of the volume.
 - III. Select the same Availability Zone as the instance.
 - IV. On **Snapshot** select the latest log backup.
 - V. Click **Yes, Create**.
- b. Attach the volume to the instance.
- c. Mount the file system.

NOTE: FOR SAFETY REASONS, THE FILE SYSTEM WILL BE MOUNTED TO A DIFFERENT MOUNT POINT FROM THE ARCHIVE LOG FILE SYSTEM ITSELF.

- I. Log on to the instance using PuTTY or any other SSH client.
- II. Mount the backup file system; for example:

```
vgscan
vgimport vgbackuplog
vgchange vgbackuplog -a y
mkdir /mnt/backuplogs
mount /dev/vgbackuplog/backuplogs /mnt/backuplogs
```

4. Restore and recover the database using CLI.
 - a. Log on into the instance as root.
 - b. Run the following commands:

```
# Log on as db2<sid>
su - db2<sid>

# If required, copy additional DB logs to the default log
archive location
# For example:
cp -rp /mnt/backuplogs/* /db2/<SID>/log_archive

# Start database manager
db2start

# Start the db recovery
db2 recover db <sid> using history file
(/db_data_backups/db2rhist.asc)

# Once DB recovery is done, restore configuration stored
outside of DB
cd /db_data_backups
db2cfimp <SID>_cfg_backup.txt
```

5. Start the SAP system.

```
su - <sidadm>
startsap all
```

Appendix 4: SAP MaxDB Database Backup to a Local File System

Example: Full Online Data and Log Backup for MaxDB

Make sure that the DB-DATA-BACKUPS and DB-LOG-BACKUPS file systems have enough storage space, and ensure proper read/write permissions for these file systems. The file system names listed in Table 5 will be used in this example.

In addition to the general references mentioned in the section [Prerequisite Documentation](#), specific MaxDB backup and recovery examples can also be found at [SAP MaxDB HowTo](#) on the SAP Community Network.

In this appendix, we'll discuss two options for backing up the MaxDB database:

- Option 1: Database and log backup using MaxDB Database Studio
- Option 2: Database and log backup using DBMCLI

Create MaxDB Backup Templates

1. Start SAP MaxDB Database Studio.
You can download this software from <http://www.sdn.sap.com/irj/scn/maxdb-downloads> and install it on any computer for remote management.
2. Add the server and database into the landscape (if required).
Go to: **My Landscape / Servers / (right-click) /Add/"Server/Database"**)
3. Type the server name or ip: on "Server Name:" and choose **Next**.
4. Select the databases, and choose **Finish**.
5. Log in to the database using the CONTROL user :
My Landscape / Servers / <servername/ip>/<DB>/ (right-click) /Login
6. Right-click on the database, and choose **Administration**.
7. Go to the **Backup** tab.
8. Expand **Templates**.
9. Create a backup template for **FULL** backup.
10. Choose **New**.

Name	<Template Name> can be FULL
Backup Type	COMPLETE DATA
Device Type	FILE
Backup Tool	NONE
Device/File	/db_data_backups/<SID>_FULL
Compressed	Unselect

11. Click **OK**.
12. Create a backup template for **LOG Backup**.

Name	<Template Name> can be LOG
Backup Type	LOG
Device Type	FILE
Backup Tool	NONE
Device/File	/db_log_backups/<SID>_LOG
Compressed	Unselect

Option 1: Database and Log Backup Using MaxDB Database Studio

Back up the database using Database Studio

1. Start the SAP MaxDB Database Studio.
2. Log in to the database using the CONTROL user:
My Landscape / Servers / <servername/ip>/<DB>/ (right-click) **/Login**
3. Right-click on the database, and choose **Administration**.
4. Go to the **Backup** tab.
5. Expand **Templates**.
6. Right-click on the template **FULL**, and choose **Backup**.

Back up the database transaction logs using Database Studio

1. Start the SAP MaxDB Database Studio.
2. Log in to the database using the CONTROL user:
My Landscape / Servers / <servername/ip>/<DB>/ (right-click) **/Login**
3. Right-click on the database, and choose **Administration**.
4. Go to the **Backup** tab.
5. Expand **Templates**.
6. Right-click on the template **LOG**, and choose **Backup**.

Option 2: Database and Log Backup Using DBMCLI

Back up the database using DBMCLI

1. Log in to the Amazon EC2 instance.
2. Run the following commands:

```
su - <sid>adm
#start an utility session
dbmcli -d <SID> -U c -uUTL
#start backup using the template FULL
backup_start FULL
```

3. Wait until the backup has completed.

Back up the database log using DBMCLI

1. Log in to the Amazon EC2 instance.
2. Run the following commands:

```
su - <sid>adm
#start an utility session
dbmcli -d <SID> -U c -uUTL
#start backup using the template FULL
backup_start LOG
```

3. Wait until the backup has completed.

Schedule Hourly Automatic Log Backup Using Database Studio

1. Start the SAP MaxDB Database Studio.
2. Log in to the database using the CONTROL user:
My Landscape / Servers / <servername/ip>/<DB>/ (right-click) **/Login**
3. Right-click on the database, and choose
Administration Tasks/Automatic Log Backup
4. Select the template **LOG** from the list.
5. Select **Create a log backup every <XX> minutes.**
6. Type **60** on the **<XX>** field.
7. Click **Activate.**

The Amazon EBS snapshot creation of the volume that holds the DB-LOG-BACKUPS file system can now be automated through a script, which is time-synchronized with the automatic log backup.

Create a Snapshot to Send the Backup to Amazon S3

Please refer to the section [Creating a Snapshot of a Backup Amazon EBS Volume](#) for detailed steps.

We recommend that you tag snapshots with a description, for example:

- **<SID>_MAXDB_DATA_BACKUPS_<YYYY-MM-DD>** for the database data backups
- **<SID>_MAXDB_LOG_BACKUPS_<YYYY-MM-DD-HH-MM-SS>** for the database log backups

Appendix 5: SAP MaxDB Database Restore and Recovery

The restoration and recovery of databases require careful planning and preparation. Take time for root-cause analysis so you can identify the most efficient recovery strategy before executing.

For example, you need to restore data only from Amazon S3 only if recovery is not possible any longer from data and backups that are already available on the system. If a database needs to be recovered to the latest possible point in time, always make sure not to overwrite and destroy the latest database transaction logs. These are typically not yet archived or backed up to Amazon S3, so the latest transactions contained in these logs could then get lost forever!

This guide does not intend to replace the original backup and restore documentation provided by database vendors and listed in [Prerequisite Documentation](#). Restoration and recovery scenarios are diverse, and they depend on the environment and the cause of the failure, so we strongly recommend that you follow the original documentation in case of a real failure.

Example: MaxDB Restore and Recovery from Amazon EBS Backup Volume Snapshots

1. If this is a disaster recovery, make sure that you have restored the operating system and the SAP system, either from an image backup (AMI) or from a backup, as described in [Appendix 1](#).
2. If required, mount the database backup on the Amazon EC2 instance.

NOTE: GENERALLY, THE MOST RECENT BACKUP SHOULD ALREADY BE AVAILABLE ON THE AMAZON EC2 INSTANCE.

- a. Only if required, create a new Amazon EBS volume based on an Amazon S3 snapshot backup.
 - I. On **Volumes**, click **Create Volumes**.
 - II. Specify the **Size** of the volume.
 - III. Select the same Availability Zone as the instance.
 - IV. On **Snapshot**, select the latest operating system backup.
 - V. Choose **Yes, Create**.
- b. Attach the volume to the instance.
- c. Mount the file system.
 - I. Log in to the Amazon EC2 instance.
 - II. Mount the `/db_data_backups` file system; for example, if you're using LVM2:

```
vgscan
vgimport vgbackup
vgchange vgbackup -a y
mkdir /db_data_backups
mount /dev/vgbackup/db_data /db_data_backups
```

3. If required, repeat the previous step for the `/db_log_backups` file system.

NOTE: GENERALLY, THE MOST RECENT BACKUP SHOULD ALREADY BE AVAILABLE ON THE AMAZON EC2 INSTANCE.

Option 1: Database Restore and Recovery Using MaxDB Database Studio

1. Restore the database using MaxDB Database Studio.
 - a. Log on to the instance as root.
 - b. Start `x_server`.
 - c. Log on to the database using the CONTROL user:
My Landscape / Servers / <servername/ip>/<DB>/(right-click) **/Login**
 - d. Set the database in administration mode: Right-click on the database, and then choose **Administration Tasks/Set State/Admin**.
 - e. Start the recovery: Right-click on the database, and then choose **Administration Tasks/Recovery**.
 - f. On the **Recovery of Database** window, choose **Recover a medium**.
 - g. Select **FULL** from the template list, and then choose **Next**.
 - h. Choose **Start**.
 - i. In the **Confirmation Initialization** database window, choose **OK**.
 - j. Wait until the restore has completed.

2. Apply transaction logs using Database Studio.
 - a. First restore the database using DBMCLI or Database Studio without restarting the database.
 - b. Log on to the database using the CONTROL user:
My Landscape / Servers / <servername/ip>/<DB>/(right-click) **/Login**
 - c. Start the recovery: Right-click on the database, and then choose **Administration Tasks/Recovery**.
 - d. You can choose whether you want to restore until a specific time (point-in-time recovery). In this example, we will restore until the last available log, so do not select **Recover until a specific time**. Select **Recover a Medium**, and then choose **Next**.

[NOTE: CREATE AN AMAZON EBS VOLUME FROM AN AMAZON S3 LOG BACKUP SNAPSHOT \(FOLLOWING THE LOG SEQUENCE\) AND MOUNT IT TO THE /DB LOG BACKUPS IF IT IS REQUIRED, TO APPLY OLDER LOGS AS THE ONES AVAILABLE ON THE SYSTEM. HOWEVER, BEFORE DOING SO, MAKE SURE TO HAVE A SNAPSHOT AVAILABLE OF THE LATEST CONTENT OF THE /DB LOG BACKUPS FILE SYSTEM, AS THAT MIGHT BE REQUIRED LATER IN THE RECOVERY PROCESS.](#)
 - e. Select **LOG** from the template list, and then choose **Next**.
 - f. Select the **Log** file number, choose **Next** (in this case, put the next log after the backup), and then choose **Next**.
 - g. Choose **Start**.
The recovery session will try to recover all consecutive logs until it fails with a missing log. You can choose to restore more logs and continue.
 - h. Choose **Start Database**, and then choose **Continue**.

Option 2: Database Restore and Recovery Using MaxDB DBMCLI

1. Restore the database using DBMCLI.
 - a. Log in to the Amazon EC2 instance as root.
 - b. Run the following example commands:

```
#Fix permissions
chown sdb:sdba /sapdb/<SID>/sapdata
chown sdb:sdba /sapdb/<SID>/saplog

#logon as <sid>adm
su - <sid>adm

#start the db in ADMIN mode
dbmcli -U c db_admin

#Logon a recovery session
dbmcli -U c

#List the backup history, and last logs
#take notes of the next log number

backup_history_open

backup_history_list -r last -c
label,action,pages,firstlog,lastlog,media

#If possible, check if the backup is
accessed as expected
recover_check FULL data

#Restore the database

db_connect
recover_start FULL data
```

- c. Wait until the restore is complete.
To monitor the restore session, use the following command:

```
#logon as <sid>adm
su - <sid>adm

#start restore session session
dbmcli -U c -uUTL -d <SID>
recover_state
```

Monitor the **Pages Transferred** and **Pages Left** values.

2. Restore the database logs using DBMCLI.
 - a. Before restoring logs, you should have restored a database without restarting (see step 1 or step 3). The database should be in ADMIN mode.
 - b. Log on to the instance as root.

- c. Run the following commands:

```
su - <sid>adm

#Logon a recovery session
dbmcli -U c

#List the backup history, and last logs
#take notes of the next log number

backup_history_open

backup_history_list -r last -c
label,action,pages,firstlog,lastlog,media

service_connect

#Restore the Logs where <XXX> is the next log sequence.
db_connect
recover_start LOG log <XXX>

#If the recovery ends with -8020 error code and you still
have logs to recover that are not listed in the backup
history, you can continue with the following commands,
where <YYY> is the next log to recover, recover log by log
until you restore the latest available log.

recover_replace LOG /backuplog/<SID>_LOG.<YYY>

#Use the following command to review the status of the
database after the restore

db_res:tartinfo

#If the consistent=1 the database can start
```

- d. Use the following commands to start the recovery if you want to restore in point-in-time recovery:

```
su - <sid>adm

#Logon a recovery session
dbmcli -U c

#List the backup history, and last logs
#take notes of the next log number

backup_history_open

backup_history_list -r last -c
label,action,pages,firstlog,lastlog,media

service_connect

#Restore the Logs where <XXX> is the next log sequence.
```

```
db_connect
recover_start LOG log <XXX> UNTIL <date> <time>
#If the recovery ends with -8020 error code and you still
have logs to recover that are not listed in the backup
history, you can continue with the following commands,
where <YYY> is the next log to recover, recover log by log
until you restore the latest available log.

recover_replace LOG /backuplog/<SID>_LOG.<YYY>

#Note: the database will be put on ONLINE automatically
after the DB is recovered until the time specified
```

3. Start the SAP instance:

```
su - <sidadm>
startsap all
```

At this point, the SAP instance should start with no issues.

4. After restoring the database, you can remove the backup Amazon EBS volume if required.
- Log on to the Amazon EC2 instance.
 - Dismount and remove the volume; for example:

```
umount /db_data_backups
vgchange vgbackup -a n
vgexport vgbackup
```

- Log on to the Amazon EC2 Management Console at <https://console.aws.amazon.com/ec2/>.
- Go to **Volumes**, and select the volume to remove.
- Choose **Detach Volume**.
- Choose **Yes, Detach** in the popup window.
- When the volume has been detached, click **Delete Volume**.

Appendix 6: Sybase ASE Database Backup to a Local File System

Sybase ASE provides two methods for backing up databases.

- Option 1: Online database backups with the **DUMP DATABASE** command.
- Option 2: External backups while the database has been quiesced; that is, while write access to the database has been suspended.

The following sections describe only option 1. For more information on external backups, see the Sybase ASE documentation.

Ensuring Recoverability of the Sybase ASE Database Server

To ensure recoverability of the Sybase ASE server, you need to be able to restore a current backup of the master database, a current backup of the <DBSID> database, and the complete sequence of the transaction log archives that have been taken since the last database backup.

For **production systems**, you **must** archive the transaction log of the <DBSID> database. The following database options must be set for the <DBSID> database:

- Set **trunc log on chkpt** to **false**
- Set **full logging for all** to **true**
- Set **enforce dump tran sequence** to **true**

For more information, refer to [SAP Note 1585981](#).

Preparation

When dumping a database or a transaction log to a disk device, it is important not to overwrite existing dumps and to keep track of the sequence. We strongly recommend that you generate device names that include a timestamp. Ideally, you call the **DUMP** command from within a stored procedure that generates a device file name, which will include the current timestamp and database name. Then you schedule calls to these stored procedures in Sybase ASE Job Scheduler. Additionally, we suggest defining a threshold that triggers a dump of the transaction log when the log fills. [SAP Note 1588316](#) provides instructions how to achieve this.

- Create the stored procedures `sp_dumpdb` and `sp_dumptrans` to call the dump command. [SAP Note 1588316](#) provides templates for these procedures. Replace the path as appropriate, use `/db_log_backups` as the directory for transaction log dumps. Use `/db_data_backups` as the directory for database dumps.
- Create the stored procedure `sp_thresholdaction` in the <DBSID> database (see [SAP Note 1588316](#)).

Automating Database and Transaction Log Dumps

Sybase ASE provides an integrated job scheduler, which you may use to automate database and transaction log dumps. The Job Scheduler should be set up by default in your SAP system.

Create a job and a schedule for each database you want to back up. Open the transaction DBACOCKPIT, and then switch to the tab **Jobs** → **DBA Planning Calendar**.

Create a job to execute stored procedure `sp_dumpdb` for the databases `master`, `<DBSID>`, `saptools`, and `sybmgmtdb` (the latter two are optional). We recommend that you schedule these jobs at least daily.

Create a job to call store procedure `sp_dumptrans` for the `<DBSID>` database. In order to minimize the potential for transaction loss in the case of a disk error, we recommend that you schedule this job at least hourly in a production environment.

Appendix 7: Sybase ASE Server Recovery

A complete restoration and recovery of a Sybase ASE server for an SAP system consists of these tasks:

1. Recreate the ASE software installation from a file system copy.
2. Recreate the ASE server.
3. Load the `master` database.
4. Load the `sybmgmt` database (optional).
5. Load the `saptools` database (optional).
6. Load the `<DBSID>` database.
7. Load the sequence of transaction log dumps of the `<DBSID>` database.

Detailed instructions can be found in [SAP Note 1618817](#) (Unix) and [SAP Note 1611715](#) (Windows). You should also refer to the [Sybase ASE documentation](#) in the Sybase Infocenter.

Particularly for production systems, we strongly recommend that you perform test restores at regular intervals to validate your restore process.

© 2016, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. 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.