
AWS Prescriptive Guidance

Evaluate downgrading Microsoft SQL Server from Enterprise edition to Standard edition on AWS



AWS Prescriptive Guidance: Evaluate downgrading Microsoft SQL Server from Enterprise edition to Standard edition on AWS

Copyright © 2022 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Introduction	1
Overview	1
Targeted business outcomes	1
Prerequisites	2
Prerequisites	2
Limitations	2
Product versions	2
Tools	3
Assessing your environment	4
Using a T-SQL script	4
Using AWS SCT	5
Scaling the SQL Server assessment	5
Compare features	6
Best practices	8
Best practices for downgrading SQL Server Enterprise edition to Standard edition	8
Best practices for Amazon RDS for SQL Server	8
FAQ	9
Can I change the Amazon RDS for SQL Server edition that I'm running for a DB instance?	9
Which Microsoft SQL Server database editions are available with Amazon RDS for SQL Server?	9
If I change the Amazon RDS for SQL Server edition, do I need new licenses?	9
What types of licensing options are available?	9
What are the licensing policies for using Amazon RDS for SQL Server?	10
How does the license option impact DB instance scaling?	10
How can I migrate from an on-premises or an Amazon EC2 Enterprise edition of SQL Server to Standard edition on AWS?	10
How can I reduce downtime when downgrading?	10
Can I have high availability?	10
How can I monitor performance?	11
Can I have encryption?	11
How is Amazon RDS for SQL Server supported?	11
How do I know if Amazon RDS supports a specific SQL Server database feature?	11
Next steps	12
Downgrading with downtime	12
Downgrading with reduced downtime	12
Resources	13
Document history	14
Glossary	15
Migration terms	15

Evaluate downgrading Microsoft SQL Server from Enterprise edition to Standard edition on AWS

Marcelo Fernandes and Paurav Chudasama, Amazon Web Services (AWS)

July 2021 ([document history](#) (p. 14))

This guide discusses how to perform an in-depth assessment of Microsoft SQL Server Enterprise edition databases and convert them to SQL Server Standard edition during a migration to [Amazon Relational Database Service \(Amazon RDS\)](#) on Amazon Web Services (AWS). The information presented also applies if your SQL Server Enterprise edition databases are already running on Amazon RDS.

This guide is for program or project managers, product owners, database and application administrators, database engineers, operations or infrastructure managers, and database and system architects.

Overview

In many organizations, SQL Server Enterprise edition is the default deployment option for production workloads. This might be due to a common misconception that a SQL Server Standard edition database is not capable of supporting enterprise-class applications. Both Enterprise edition and Standard edition share a common code base, so technically they offer the same SQL Server database functionalities.

SQL Server Enterprise edition offers additional options, but it is significantly more expensive than the Standard edition. Downgrading to Standard edition provides an opportunity to reduce the overall total cost of ownership of your databases. Applications with minimum or no usage of Enterprise edition features are good candidates for downgrades to SQL Server Standard edition.

Amazon RDS facilitates setting up, operating, and scaling a relational database in the cloud. It provides cost-efficient and [resizable capacity](#) while automating time-consuming administration tasks such as hardware provisioning, database setup, patching, and backups. Amazon RDS for SQL Server offers the License Included (LI) model. For more details, see [Amazon RDS for SQL Server FAQs](#).

SQL Server Standard edition on Amazon RDS delivers enterprise-class features and performance with options such as [Multi-AZ deployment](#) for high availability, [Performance Insights](#) for performance tuning, and [Amazon CloudWatch](#) for monitoring.

Targeted business outcomes

The goal of this guide is to provide the following business outcomes:

- **Cost savings on SQL Server licensing fees** – Downgrading to Standard edition reduces the total cost of running your applications.
- **Reduce the need for large capital outlays** – Amazon RDS for SQL Server offers License Included (LI) licensing, which allows you to pay only for what you use.
- **Efficient assessment** – Bulk assessment of your database estate helps in determining the suitability of downgrading to Standard edition.

Prerequisites and limitations

Prerequisites

- No application support requirements for SQL Server Enterprise edition
- An active AWS account
- Secure network connectivity, through a virtual private network or [AWS Direct Connect](#), between your on-premises data center and a virtual private cloud (VPC) on AWS
- SQL Server Enterprise edition running in an on-premises data center or on an Amazon Elastic Compute Cloud (Amazon EC2) instance or on Amazon RDS for SQL Server
- A database client tool, such as SQL Server Management Studio (SSMS), for running Transact-SQL (T-SQL) commands
- Database user account with appropriate permissions for running AWS Schema Conversion Tool (AWS SCT) assessment

Limitations

- Amazon RDS for SQL Server has storage size and IOPs limits. For the current maximum, see the [AWS documentation](#).
- AWS SCT supports SQL Server version 2008 and later.

Product versions

The general logic described in this guide applies to SQL Server versions from 2005 and later. However, AWS SCT supports only SQL Server versions 2008 and later. To identify feature usage in cases where AWS SCT is not supported, run SQL queries on the source database.

For a current list of supported versions and editions, see [Microsoft SQL Server on Amazon RDS](#) in the AWS documentation. For details on pricing and supported instance classes, see [Amazon RDS for SQL Server pricing](#).

Tools

- [AWS SCT](#) – AWS Schema Conversion Tool (AWS SCT) provides a project-based user interface to automatically assess, convert, and copy the database schema of your source SQL Server database into a format that is compatible with Amazon RDS for SQL Server. Using AWS SCT, you can analyze the potential cost savings that you can achieve by changing your license type from SQL Server Enterprise edition to Standard edition.
- [SSMS](#) – Microsoft SQL Server Management Studio (SSMS) is an integrated environment for managing SQL Server infrastructure. It provides a user interface and a group of tools with rich script editors that interact with SQL Server.

Assessing your environment

To assess your SQL Server environment and find out whether your Enterprise SQL Server instance is using Enterprise feature, you can take one of the following approaches:

- Using a T-SQL script
- Using the AWS SCT tool

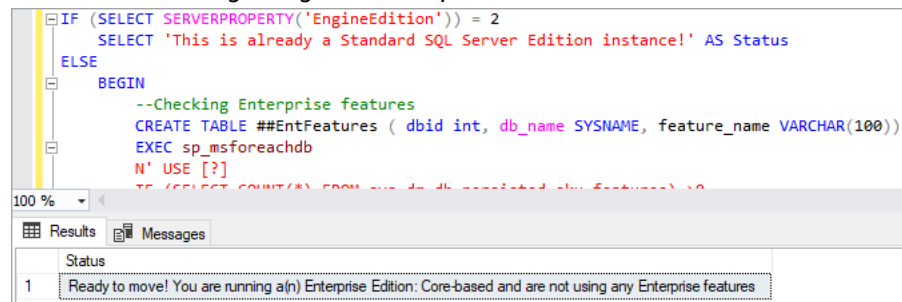
Using a T-SQL script to assess a SQL Server database

To determine whether your Microsoft SQL Server database uses Enterprise edition-specific features, you can run the following T-SQL statement in SQL Server Management Studio (SSMS).

```
IF (SELECT SERVERPROPERTY('EngineEdition')) = 2
    SELECT 'This is already a Standard SQL Server Edition instance!' AS Status
ELSE
    BEGIN
        --Checking Enterprise features
        CREATE TABLE ##EntFeatures ( dbid int, db_name SYSNAME, feature_name VARCHAR(100))
        EXEC sp_msforeachdb
            N' USE [?];
            IF (SELECT COUNT(*) FROM sys.dm_db_persisted_sku_features) >0
            BEGIN
                INSERT INTO ##EntFeatures
                    SELECT db_id(),dbname=DB_NAME(),feature_name FROM
            sys.dm_db_persisted_sku_features
            END '
        IF (SELECT COUNT(1) FROM ##EntFeatures WHERE dbid>4) >0
            SELECT db_name,feature_name FROM ##EntFeatures;
        ELSE
            SELECT CONCAT('Ready to move! You are running a(n) ',
            CONVERT(VARCHAR,SERVERPROPERTY('Edition')), ' and are not using any Enterprise features')
            AS Status
        END
    END

IF OBJECT_ID('tempdb.dbo.##EntFeatures') IS NOT NULL
    DROP TABLE ##EntFeatures
```

If the script returns a single row with the message **Ready to move! You are running a(n) <your SQL Server edition>**, as shown in the following screen illustration, your SQL Server database can be a candidate for downgrading from Enterprise edition to Standard edition.



AWS Prescriptive Guidance Evaluate
downgrading Microsoft SQL Server from
Enterprise edition to Standard edition on AWS
Using AWS SCT

If the script returns a list of databases and Enterprise edition features, you must evaluate the possible consequences of removing Enterprise features from those databases.

```
IF (SELECT SERVERPROPERTY('EngineEdition')) = 2
    SELECT 'This is already a Standard SQL Server Edition instance!' AS Status
ELSE
    BEGIN
        --Checking Enterprise features
        CREATE TABLE ##EntFeatures ( dbid int, db_name SYSNAME, feature_name VARCHAR(100))
        EXEC sp_msforeachdb
            N' USE [%]
            IF (SELECT COUNT(*) FROM sys.dm_server_registry_values WHERE ...)
```

db_name	feature_name
1 AdventureWorks2016	Partitioning
2 WideWorldImporters	Compression
3 WideWorldImporters	Partitioning
4 WideWorldImporters	ColumnStoreIndex
5 WideWorldImporters	InMemoryOLTP

Using AWS SCT to assess a SQL Server database

When you create a database migration assessment report in AWS SCT, the **License Evaluation and Cloud Support** section of the report lists the SQL Server features that your source database is using. You can use this list to make an informed decision while migrating to Amazon RDS for SQL Server.

License evaluation and cloud support
Your Enterprise server has the following features turned on: Compression.
If you want to move this database to Standard Edition, you need to remove dependencies on these features: ColumnStoreIndex, Partitioning.
Your source server has some features turned on that can't be moved to RDS: InMemoryOLTP.

Scaling the SQL Server assessment

AWS SCT now supports batch assessment using the multiserver assessor to evaluate multiple servers based on input that you provide for each schema definition that you want to assess. Your schema definition contains database server connection parameters and the full name of each schema. After assessing each schema, the assessor produces a summary aggregated report that shows the estimated complexity for each possible migration target. It also produces a server-level report that includes the **License Evaluation and Cloud Support** section.

Compare SQL Server Enterprise and Standard edition features

The following table is a short version of the list of Microsoft SQL Server Enterprise edition features and possible alternatives in the Standard edition. For additional information, see a more extensive [feature comparison](#).

Functional area	Feature	Enterprise edition	Standard edition	Notes
Scale limits	Maximum number of processors or cores per instance	OS maximum	4 sockets or 24 cores	See DB instance class support for SQL Server on Amazon RDS for SQL .
	Maximum memory used per instance	OS maximum	128 GB (SQL Server 2016 and later)	
	Maximum memory-optimized data size per database	Unlimited memory (SQL Server 2014 and later)	32 GB (SQL Server 2016 and later)	
High availability	Always On availability groups	Yes	Basic availability groups (SQL Server 2016 and later)	The Multi-AZ feature of Amazon RDS replicates database updates across two Availability Zones to increase durability and availability. For planned maintenance and unplanned disruptions, Amazon RDS automatically fails over to the standby.
	Online indexing	Yes	No	Where possible, plan for these activities during a downtime window.
	Online schema change	Yes	No	
	Resumable online index rebuilds	Yes (SQL Server 2017 and later)	No	
	Accelerated database recovery (ADR)	Yes (SQL Server 2019)	Yes (SQL Server 2019)	Both SQL Server editions support ADR in SQL Server 2019.

AWS Prescriptive Guidance Evaluate
downgrading Microsoft SQL Server from
Enterprise edition to Standard edition on AWS

Functional area	Feature	Enterprise edition	Standard edition	Notes
Scalability and performance	Table and index partitioning	Yes	Yes (SQL Server 2016 SP1 and later)	
	Data compression	Yes	Yes (SQL Server 2016 SP1 and later)	
	Partition table parallelism	Yes	Yes (SQL Server 2016 SP1 and later)	
	In-Memory OLTP	Yes (SQL 2014 and later)	Yes (SQL Server 2016 SP1 and later)	
	Delayed durability	Yes (SQL 2014 and later)	Yes (SQL 2014 and later)	
	Columnstore	Yes (SQL Server 2016 SP1 and later)	Yes (SQL Server 2016 SP1 and later)	
	Interleaved execution for multi-statement table valued functions	Yes (SQL Server 2017 and later)	Yes (SQL Server 2017 and later)	
	Intelligent Database: batch mode on rowstore	Yes (SQL Server 2019)	No	Not available in Standard edition, but you can take advantage of the other features of Intelligent Database.
	Intelligent Database: approximate count distinct	Yes (SQL Server 2019)	Yes (SQL Server 2019)	
	Intelligent Database: table variable deferred compilation	Yes (SQL Server 2019)	Yes (SQL Server 2019)	
	Intelligent Database: scalar UDF inlining	Yes (SQL Server 2019)	Yes (SQL Server 2019)	
Security	Transparent database encryption (TDE)	Yes	Yes (SQL Server 2019)	
	Contained database	Yes	Yes	

Best practices

Best practices for downgrading SQL Server Enterprise edition to Standard edition

The following best practices are recommended for assessing and downgrading your SQL Server Enterprise edition to Standard edition on Amazon RDS for SQL Server.

- **Validate vendor support for SQL Server Standard edition** – Ensure that your application is certified to run on SQL Server Standard edition. Check directly with the software vendor or their documentation. Downgrading to Standard edition without vendor confirmation might invalidate support for the application.
- **Identify the use of Enterprise edition features for administration activities** – Database administrators or application administrators sometimes rely on Enterprise edition features to improve operational processes. Some examples are online maintenance activities (such as online index create and rebuild). Identify usage of such features and make an informed decision. You can modify your operations or apply workarounds where possible to mitigate the loss of Enterprise edition features.
- **Assess workload patterns of SQL Server Enterprise edition** – SQL Server Standard edition automatically restricts usage to a maximum of 24 CPU cores or 4 sockets at any time. For a list of instance types that are available for SQL Server Standard edition, see the [Amazon RDS for SQL Server documentation](#). Analyze your database workload pattern to determine whether the maximum limit of 24 CPU cores will negatively impact service levels when you downgrade to Standard edition.
- **Test your workload against SQL Server Standard edition** – Run multiple tests using representative workload to determine whether the 24-core limit imposed by Standard edition will have a negative impact on your workload. Monitor results and tune the database as required.
- **Test database administration activities** – Monitoring, performance assessments, tuning, and maintenance tasks to build comfort levels around new ways of working in the Standard edition environment.
- **Test the migration process** – Run comprehensive migration tests to build operator confidence, fine-tune the approach, and document every issue with resolution.

Best practices for Amazon RDS for SQL Server

For best practices for working with Amazon RDS for SQL Server, see the [User Guide](#) and view the [Deep dive on new features in Amazon RDS for SQL Server video](#).

FAQ

This section provides answers to commonly raised questions about downgrading from SQL Server Enterprise edition to Standard edition and using Amazon RDS for SQL Server.

Can I change the Amazon RDS for SQL Server edition that I'm running for a DB instance (for example, from SQL Server 2016 Standard edition to Enterprise edition)?

Yes, to change the edition and retain your data, take a snapshot of your running DB instance, and create a DB instance of the desired edition from that snapshot. Then delete the old DB instance, unless you want to keep it running.

Which Microsoft SQL Server database editions are available with Amazon RDS for SQL Server?

Amazon RDS supports Microsoft SQL Server versions 2012–2019 and the following editions:

- Enterprise
- Standard
- Web
- Express

For more information about supported versions and editions, see [Microsoft SQL Server on Amazon RDS](#).

If I change the Amazon RDS for SQL Server edition (for example, change from Enterprise edition to Standard edition), do I need to buy new licenses?

When you set up an Amazon RDS DB instance for Microsoft SQL Server, the software license is included. This means that you don't need to purchase SQL Server licenses separately. AWS holds the license for the SQL Server database software.

What types of licensing options are available with Amazon RDS for SQL Server?

When you set up an Amazon RDS DB instance for Microsoft SQL Server, the software license is included. This means that you don't need to purchase SQL Server licenses separately. AWS holds the license for the

What are the licensing policies for using Amazon RDS for SQL Server?

Amazon RDS for SQL Server uses the License Included service model, so you do not need separately purchased SQL Server licenses. The SQL Server database software has been licensed by AWS for your use subject to Section 10.4 of the [AWS Service Terms](#).

How does the license option impact DB instance scaling?

DB instances running SQL Server can be scaled up and down at any point, subject to the prevailing hourly pricing for each DB instance class.

How can I migrate from an on-premises or an Amazon EC2 Enterprise edition of SQL Server to Standard edition on AWS?

Migrating from Enterprise edition to Standard edition requires logical export of data from the Enterprise edition database and importing the data into the Standard edition database instance. To move your data, use [backup and restore](#) or [AWS Database Migration Service \(AWS DMS\)](#).

How can I reduce downtime when downgrading my Enterprise edition SQL Server database to Standard edition?

To reduce overall downtime, bulk load data into your Standard edition database. You can use backup and restore and logical replication tools such as [AWS DMS](#) to synchronize your Standard edition database.

Can I have high availability in my Standard edition of Amazon RDS for SQL Server?

Amazon RDS currently uses synchronous replication technology and automatic failover functionality to provide Multi-AZ deployments for SQL Server DB instances. Multi-AZ deployments are available for SQL Server Standard and Enterprise editions. For more information, see [Multi-AZ deployments for Amazon RDS for Microsoft SQL Server](#).

How can I monitor the performance of my Amazon RDS for SQL Server Standard edition DB instance?

To monitor your Standard edition DB instance, you can use [Amazon RDS Performance Insights](#), [Enhanced Monitoring](#), and [Amazon CloudWatch](#).

Can I have encryption on my Amazon RDS for SQL Server Standard edition DB instance?

Amazon RDS can encrypt your Amazon RDS DB instances. Data that is *encrypted at rest* includes the underlying storage for DB instances, its automated backups, read replicas, and snapshots. To enable *encryption in transit*, you can use [Secure Sockets Layer \(SSL\) encryption for a SQL Server DB instance](#) with the `rds.force_ssl` parameter.

How is Amazon RDS for SQL Server supported?

If you have an active AWS Support account, you can contact AWS Support for service requests relating to Amazon RDS or SQL Server.

How do I know if Amazon RDS supports a specific SQL Server database feature?

The SQL Server features that Amazon RDS for SQL Server supports vary, depending on the edition of SQL Server. For information about the SQL Server features that Amazon RDS currently supports, see the [Amazon RDS User Guide](#).

Next steps

After you determine that you can safely downgrade an instance from SQL Server Enterprise edition, the next step is to migrate to SQL Server Standard edition on Amazon RDS. To migrate the database, use one or a combination of the following tools:

- **Native backup and restore** – You can use native backup and restore to move data in and out of SQL Server database instances. Amazon RDS supports [native backup and restore](#) for Microsoft SQL Server databases using full backup (.bak) files. When you use Amazon RDS, you access files stored in Amazon Simple Storage Service (Amazon S3) rather than using the local file system on the database server.
- **AWS Database Migration Service (AWS DMS)** – AWS DMS helps you migrate relational databases, data warehouses, and other types of data stores. You can use AWS DMS to migrate your data into the AWS Cloud or between combinations of cloud and on-premises databases. The Change Data Capture option of AWS DMS offers continuous replication, so you can reduce the total downtime during migration. For information about SQL Server versions and editions that AWS DMS supports, see the [AWS DMS documentation](#).

Depending on your availability requirements during the downgrade, you can adopt any of the following options to safely downgrade your Enterprise edition instance to a Standard edition on Amazon RDS for SQL Server.

Downgrading with downtime

- Use [native backup and restore](#) to create a consistent copy from the Enterprise edition database and restoring to the Standard edition database.
- Use [AWS DMS](#) to perform a full load of data from the Enterprise edition to the Standard edition database.

Downgrading with reduced downtime

- Use [AWS DMS](#) for full load and data synchronization.

Resources

References

- [Amazon RDS for SQL Server features](#)
- [Best Practices for Deploying Microsoft SQL Server on AWS](#)
- [Amazon RDS for SQL Server User Guide](#)
- [Migrate a self-managed Microsoft SQL Server database to a fully managed database on Amazon RDS](#)

Video

- [Deep Dive on new features in Amazon RDS for SQL Server \(re:Invent 2019\)](#)

Tools

- [AWS Schema Conversion Tool \(AWS SCT\)](#)
- [AWS Database Migration Service \(AWS DMS\)](#)

AWS Prescriptive Guidance pattern and guide

- [Migrate an on-premises Microsoft SQL Server database to Amazon RDS for SQL Server](#)
- [Migrating Microsoft SQL Server databases to the AWS Cloud](#)
- [Choosing between Amazon EC2 and Amazon RDS](#)

AWS Partners

- [AWS Microsoft Workloads Competency Partners](#)

Document history

The following table describes significant changes to this guide. If you want to be notified about future updates, you can subscribe to an [RSS feed](#).

Change	Description	Date
– (p. 14)	Updated the answer to the FAQ about high availability .	July 14, 2022
– (p. 14)	Updated values in the Compare SQL Server Enterprise and Standard edition features section.	April 22, 2022
– (p. 14)	Initial publication	July 12, 2021

AWS Prescriptive Guidance glossary

The following are commonly used terms in strategies, guides, and patterns provided by AWS Prescriptive Guidance. To suggest entries, please use the **Provide feedback** link at the end of the glossary.

Migration terms

7 Rs

Seven common migration strategies for moving applications to the cloud. These strategies build upon the 5 Rs that Gartner identified in 2011 and consist of the following:

- Refactor/re-architect – Move an application and modify its architecture by taking full advantage of cloud-native features to improve agility, performance, and scalability. This typically involves porting the operating system and database. Example: Migrate your on-premises Oracle database to the Amazon Aurora PostgreSQL-Compatible Edition.
- Replatform (lift and reshape) – Move an application to the cloud, and introduce some level of optimization to take advantage of cloud capabilities. Example: Migrate your on-premises Oracle database to Amazon Relational Database Service (Amazon RDS) for Oracle in the AWS Cloud.
- Repurchase (drop and shop) – Switch to a different product, typically by moving from a traditional license to a SaaS model. Example: Migrate your customer relationship management (CRM) system to Salesforce.com.
- Rehost (lift and shift) – Move an application to the cloud without making any changes to take advantage of cloud capabilities. Example: Migrate your on-premises Oracle database to Oracle on an EC2 instance in the AWS Cloud.
- Relocate (hypervisor-level lift and shift) – Move infrastructure to the cloud without purchasing new hardware, rewriting applications, or modifying your existing operations. This migration scenario is specific to VMware Cloud on AWS, which supports virtual machine (VM) compatibility and workload portability between your on-premises environment and AWS. You can use the VMware Cloud Foundation technologies from your on-premises data centers when you migrate your infrastructure to VMware Cloud on AWS. Example: Relocate the hypervisor hosting your Oracle database to VMware Cloud on AWS.
- Retain (revisit) – Keep applications in your source environment. These might include applications that require major refactoring, and you want to postpone that work until a later time, and legacy applications that you want to retain, because there's no business justification for migrating them.
- Retire – Decommission or remove applications that are no longer needed in your source environment.

application portfolio

A collection of detailed information about each application used by an organization, including the cost to build and maintain the application, and its business value. This information is key to [the portfolio discovery and analysis process](#) and helps identify and prioritize the applications to be migrated, modernized, and optimized.

artificial intelligence operations (AIOps)

The process of using machine learning techniques to solve operational problems, reduce operational incidents and human intervention, and increase service quality. For more information about how AIOps is used in the AWS migration strategy, see the [operations integration guide](#).

AWS Cloud Adoption Framework (AWS CAF)

A framework of guidelines and best practices from AWS to help organizations develop an efficient and effective plan to move successfully to the cloud. AWS CAF organizes guidance into six focus areas called perspectives: business, people, governance, platform, security, and operations. The business, people, and governance perspectives focus on business skills and processes; the platform, security, and operations perspectives focus on technical skills and processes. For example, the people perspective targets stakeholders who handle human resources (HR), staffing functions, and people management. For this perspective, AWS CAF provides guidance for people development, training, and communications to help ready the organization for successful cloud adoption. For more information, see the [AWS CAF website](#) and the [AWS CAF whitepaper](#).

AWS landing zone

A landing zone is a well-architected, multi-account AWS environment that is scalable and secure. This is a starting point from which your organizations can quickly launch and deploy workloads and applications with confidence in their security and infrastructure environment. For more information about landing zones, see [Setting up a secure and scalable multi-account AWS environment](#).

AWS Workload Qualification Framework (AWS WQF)

A tool that evaluates database migration workloads, recommends migration strategies, and provides work estimates. AWS WQF is included with AWS Schema Conversion Tool (AWS SCT). It analyzes database schemas and code objects, application code, dependencies, and performance characteristics, and provides assessment reports.

business continuity planning (BCP)

A plan that addresses the potential impact of a disruptive event, such as a large-scale migration, on operations and enables a business to resume operations quickly.

Cloud Center of Excellence (CCoE)

A multi-disciplinary team that drives cloud adoption efforts across an organization, including developing cloud best practices, mobilizing resources, establishing migration timelines, and leading the organization through large-scale transformations. For more information, see the [CCoE posts](#) on the AWS Cloud Enterprise Strategy Blog.

cloud stages of adoption

The four phases that organizations typically go through when they migrate to the AWS Cloud:

- Project – Running a few cloud-related projects for proof of concept and learning purposes
- Foundation – Making foundational investments to scale your cloud adoption (e.g., creating a landing zone, defining a CCoE, establishing an operations model)
- Migration – Migrating individual applications
- Re-invention – Optimizing products and services, and innovating in the cloud

These stages were defined by Stephen Orban in the blog post [The Journey Toward Cloud-First & the Stages of Adoption](#) on the AWS Cloud Enterprise Strategy blog. For information about how they relate to the AWS migration strategy, see the [migration readiness guide](#).

configuration management database (CMDB)

A database that contains information about a company's hardware and software products, configurations, and inter-dependencies. You typically use data from a CMDB in the portfolio discovery and analysis stage of migration.

epic

In agile methodologies, functional categories that help organize and prioritize your work. Epics provide a high-level description of requirements and implementation tasks. For example, AWS CAF security epics include identity and access management, detective controls, infrastructure security,

data protection, and incident response. For more information about epics in the AWS migration strategy, see the [program implementation guide](#).

heterogeneous database migration

Migrating your source database to a target database that uses a different database engine (for example, Oracle to Amazon Aurora). Heterogeneous migration is typically part of a re-architecting effort, and converting the schema can be a complex task. [AWS provides AWS SCT](#) that helps with schema conversions.

homogeneous database migration

Migrating your source database to a target database that shares the same database engine (for example, Microsoft SQL Server to Amazon RDS for SQL Server). Homogeneous migration is typically part of a rehosting or replatforming effort. You can use native database utilities to migrate the schema.

idle application

An application that has an average CPU and memory usage between 5 and 20 percent over a period of 90 days. In a migration project, it is common to retire these applications or retain them on premises.

IT information library (ITIL)

A set of best practices for delivering IT services and aligning these services with business requirements. ITIL provides the foundation for ITSM.

IT service management (ITSM)

Activities associated with designing, implementing, managing, and supporting IT services for an organization. For information about integrating cloud operations with ITSM tools, see the [operations integration guide](#).

large migration

A migration of 300 or more servers.

Migration Acceleration Program (MAP)

An AWS program that provides consulting support, training, and services to help organizations build a strong operational foundation for moving to the cloud, and to help offset the initial cost of migrations. MAP includes a migration methodology for executing legacy migrations in a methodical way and a set of tools to automate and accelerate common migration scenarios.

Migration Portfolio Assessment (MPA)

An online tool that provides information for validating the business case for migrating to the AWS Cloud. MPA provides detailed portfolio assessment (server right-sizing, pricing, TCO comparisons, migration cost analysis) as well as migration planning (application data analysis and data collection, application grouping, migration prioritization, and wave planning). The [MPA tool](#) (requires login) is available free of charge to all AWS consultants and APN Partner consultants.

Migration Readiness Assessment (MRA)

The process of gaining insights about an organization's cloud readiness status, identifying strengths and weaknesses, and building an action plan to close identified gaps, using the AWS CAF. For more information, see the [migration readiness guide](#). MRA is the first phase of the [AWS migration strategy](#).

migration at scale

The process of moving the majority of the application portfolio to the cloud in waves, with more applications moved at a faster rate in each wave. This phase uses the best practices and lessons learned from the earlier phases to implement a *migration factory* of teams, tools, and processes to streamline the migration of workloads through automation and agile delivery. This is the third phase of the [AWS migration strategy](#).

migration factory

Cross-functional teams that streamline the migration of workloads through automated, agile approaches. Migration factory teams typically include operations, business analysts and owners, migration engineers, developers, and DevOps professionals working in sprints. Between 20 and 50 percent of an enterprise application portfolio consists of repeated patterns that can be optimized by a factory approach. For more information, see the [discussion of migration factories](#) and the [Cloud Migration Factory guide](#) in this content set.

migration metadata

The information about the application and server that is needed to complete the migration. Each migration pattern requires a different set of migration metadata. Examples of migration metadata include the target subnet, security group, and AWS account.

migration pattern

A repeatable migration task that details the migration strategy, the migration destination, and the migration application or service used. Example: Rehost migration to Amazon EC2 with AWS Application Migration Service.

migration strategy

The approach used to migrate a workload to the AWS Cloud. For more information, see the [7 Rs \(p. 15\)](#) entry in this glossary and see [Mobilize your organization to accelerate large-scale migrations](#).

operational-level agreement (OLA)

An agreement that clarifies what functional IT groups promise to deliver to each other, to support a service-level agreement (SLA).

operations integration (OI)

The process of modernizing operations in the cloud, which involves readiness planning, automation, and integration. For more information, see the [operations integration guide](#).

organizational change management (OCM)

A framework for managing major, disruptive business transformations from a people, culture, and leadership perspective. OCM helps organizations prepare for, and transition to, new systems and strategies by accelerating change adoption, addressing transitional issues, and driving cultural and organizational changes. In the AWS migration strategy, this framework is called *people acceleration*, because of the speed of change required in cloud adoption projects. For more information, see the [OCM guide](#).

playbook

A set of predefined steps that capture the work associated with migrations, such as delivering core operations functions in the cloud. A playbook can take the form of scripts, automated runbooks, or a summary of processes or steps required to operate your modernized environment.

portfolio assessment

A process of discovering, analyzing, and prioritizing the application portfolio in order to plan the migration. For more information, see [Evaluating migration readiness](#).

responsible, accountable, consulted, informed (RACI) matrix

A matrix that defines and assigns roles and responsibilities in a project. For example, you can create a RACI to define security control ownership or to identify roles and responsibilities for specific tasks in a migration project.

runbook

A set of manual or automated procedures required to perform a specific task. These are typically built to streamline repetitive operations or procedures with high error rates.

service-level agreement (SLA)

An agreement that clarifies what an IT team promises to deliver to their customers, such as service uptime and performance.

task list

A tool that is used to track progress through a runbook. A task list contains an overview of the runbook and a list of general tasks to be completed. For each general task, it includes the estimated amount of time required, the owner, and the progress.

workstream

Functional groups in a migration project that are responsible for a specific set of tasks. Each workstream is independent but supports the other workstreams in the project. For example, the portfolio workstream is responsible for prioritizing applications, wave planning, and collecting migration metadata. The portfolio workstream delivers these assets to the migration workstream, which then migrates the servers and applications.

zombie application

An application that has an average CPU and memory usage below 5 percent. In a migration project, it is common to retire these applications.