

[Home](#) > [Architecting](#)

Everything you should know before moving workloads to the Cloud

18 June 2025 - 1 min. read

[Application Modernization](#)

[Cloud Migration](#)

"Change is the law of life. And those who look only to the past or present are certain to miss the future."

John F. Kennedy

"The secret of change is to focus all of your energy not on fighting the old, but on building the new."

Dan Millman, Way of the Peaceful Warrior

Migrating workloads to the Cloud can be a complex process that requires significant time and focus from organizations.

In fact, it's not just about identifying what to migrate and moving it to the Cloud—it demands careful planning and some

essential preliminary activities to avoid delays, unexpected costs, and operational risks.

Over the past few years, AWS has developed and refined the **Migration Acceleration Program (MAP)**, a framework structured into three main phases — *Assess, Mobilize, Migrate, and Modernize* — designed to effectively support customers throughout the migration journey, with the help of its partner network.

Let's explore each phase in detail:



Source: <https://aws.amazon.com/migrate-modernize-build/cloud-migration/how-to-migrate/>

Even without participating in this program, you can use it as a framework to follow to start a migration.

Let's see what the phases are

The **Assess** phase aims to build a business case to evaluate the organization's readiness for Cloud Migration by estimating the licensing needs, TCO, and right-sizing for computing resources.

The **Mobilize** phase will build the migration plan using the 7Rs strategies by considering workload dependencies and refining the business case. This phase will also define the **Landing Zone**, including the **network** design.

Finally, the real magic happens in the **Migrate and Modernize** phase, when VMs are transformed into EC2 instances, Containers, RDS databases, and other Managed Services.

AWS MAP provides excellent guidelines for properly planning a migration, but that alone is not enough. Identifying the most suitable strategy for each workload is also essential.

AWS defines seven distinct strategies, known as the **7 "Rs" of migration**.

Let's take a closer look at each of them.

The 7 Rs

A key phase for a successful migration is to identify the right strategy to move workloads and applications to the Cloud.

AWS defined seven distinct strategies that can be used to identify how to migrate to the Cloud. Every strategy starts with the letter "R", hence the name.

You may ask yourself what the origin of the 7Rs is. It started in 2010 when Gartner introduced “The 5Rs migration strategy”. They originally were:

- Relocate
- Rehost
- Replatform
- Repurchase
- Refactor

AWS originally expanded this model with a new “R”: **Retire**, but then (as Cloud adoption grew), the need for another path emerged: “**Retain**”.

Everything starts with an assessment and a discovery of the workloads, using tools we’ll discuss later. A round of interviews with application and technical owners usually helps to decide the right migration strategy.

Retire

After the initial assessment, you may find a set of services and VMs that are no longer used, powered on, and left alone (it is not uncommon, trust us!). For this kind of workload, the strategy is the simplest and takes no effort: shut them down!

Decommissioning old services also positively impacts security, reducing the attack surface by eliminating old operating systems, applications, and libraries with vulnerabilities.

Retain

Some workloads cannot be migrated to the cloud for some aspects, such as data locality compliance, dependencies on physical hardware (like license dongles), for their architecture (such as Sparc, and PowerPC CPUs), or because you want to focus on migrating other applications that can bring more business value to your organization. This kind of workload can be evaluated later (and sometimes their migration strategy will be a simple “retire” again).

Rehost

This strategy is also commonly referred to as **lift and shift**. It is an infrastructural approach that leverages only the IaaS part of the AWS Cloud. You will migrate entire VMs to EC2 instances without significant modifications. With this migration path, you

can move to the Cloud quickly when external factors impose a deadline (like license renewals, hardware obsolescence, and expiring agreements with hosting/colocation providers).

You can also use this strategy to leverage Cloud elasticity to autoscale the compute platform and adapt to increased demand (be aware that the application has to be architected to scale, but this is another story). It is also easier to refactor and rearchitect applications once you can fully benefit from a Cloud environment. You can use tools like **AWS Application Migration Service** (MGN) or **VM Import/Export**. [This article](#) provides an example of a migration scenario when the service was still called CloudEndure.

Relocate

The relocate strategy moves servers and platforms to a Cloud version. This approach is usually used to migrate on-premise databases to Amazon RDS using **AWS Database Migration Service** (DMS) or to migrate entire VMware SDDCs to **VMware Cloud on AWS**, or moving Kubernetes Workloads to **Amazon EKS**. In most use cases, relocating a server doesn't impact availability or the application architecture. Mattia's article provides an excellent overview and an additional use case for [DMS](#).

Repurchase

This is the “drop and shop” approach. You can evaluate a SaaS equivalent of your on-premise application and see if it fits your business needs and requirements. This migration path is usually good for moving from a traditional license to SaaS, upgrading an old version of an application, or entirely replacing an old one.

You must carefully evaluate the time and effort required to migrate data, train the team, and integrate with existing systems. Still, you will reduce costs associated with maintenance for the application, infrastructure, and security.

Replatform

This approach (lift, tinker, and shift) introduces optimization levels while moving applications to the Cloud. It might require some changes to the application, but it lowers the effort needed to maintain the infrastructure because it focuses on using managed services.

The most common use case is containers instead of EC2 instances, [like we did in the past with WordPress](#). Be sure to [avoid common errors](#) and be listed in the next episode of our Halloween special!

Refactor

As the name suggests, refactor is the most desirable approach because you can modify your workload to be cloud-native and leverage native services. Still, you have to plan and consider the effort carefully. You will need to apply significant code changes. This migration path is also valid for databases; you can consider migrating the engine (for example, migrating from an Oracle database to a PostgreSQL database, or leveraging Babelfish to migrate away from SQL Server licensing fees). At the end of the process, you will have a shiny new cloud-native application that uses managed services, can scale, and has a low maintenance effort!

This approach is not straightforward, but luckily, we have a [series of articles](#) on this topic and an interesting use case of [API Gateway to gradually refactor an application](#).

To migrate or not to migrate...

After reviewing this long list, you may think that Cloud Migration is a project that can significantly impact your team and resources, especially for big organizations. If you have hundreds or thousands of servers, choosing the right approach for every workload and considering dependencies can be complex and error-prone.

Fear not.

Once you start analyzing and understanding your organization, the path for migration and modernization will become clear. Tools and experience will help ease the pain, and always don't fear asking questions!

Have you already started the cloud migration journey? What are your experiences with the 7Rs, and do you have a favorite? Let us know in the comments!

About Proud2beCloud

Proud2beCloud is a blog by [beSharp](#), an Italian APN Premier Consulting Partner expert in designing, implementing, and managing complex Cloud infrastructures and advanced services on AWS. Before being writers, we are Cloud Experts working daily with AWS services since 2007. We are hungry readers, innovative builders, and gem-

seekers. On Proud2beCloud, we regularly share our best AWS pro tips, configuration insights, in-depth news, tips&tricks, how-tos, and many other resources. Take part in the discussion!



Damiano Giorgi

Ex on-prem systems engineer, lazy and prone to automating boring tasks. In constant search of technological innovations and new exciting things to experience. And that's why I love Cloud Computing! At this moment, the only "hardware" I regularly dedicate myself to is that my bass; if you can't find me in the office or in the band room try at the pub or at some airport, then!



Alessio Gandini

Cloud-native Development Line Manager @ beSharp, DevOps Engineer and AWS expert. Since I was still in the Alfa version, I'm a computer geek, a computer science-addicted, and passionate about electronics all-around. At this moment, I'm hanging out in the IoT world, also exploring the voice user experience, trying to do amazing (Io)Things. Passionate about cinema and great TV series consumer, Sunday videogamer
