Considering a Mass Migration to the Cloud?

What Is a Mass Migration to the Cloud?

People have been taking advantage of the advances in technology to migrate systems to more capable platforms since the beginning of time. Hand-written books to the printing press; self-generated electricity to the power grid; human-computed encryption/decryption to the digital computer; mainframes to commodity hardware to virtualization; and so on.

The fundamental process around migration—understand the benefits of the new system, assess gaps in the existing system, plan, and migrate—hasn't changed much over time. I have found, however, that the prospect of migrating a large number of legacy applications to the cloud can sometimes be intimidating to organizations because of the sheer magnitude of change required. Modern enterprises have IT environments that become larger and more complex every day, and organizations rarely have the opportunity to retire technical debt as they continue to build new systems.

For the purposes of this mini-series, we'll consider a mass migration to be the movement of a meaningful portion of your organization's existing IT assets to the cloud, and we'll simply refer to it as a "migration." A migration might consist of a data center, a collection of data centers, a business unit, or some other portfolio of systems that is larger than a single application.

Approaching Migrations

Combining what we know about technology migrations with our experience helping organizations migrate their IT portfolios to AWS, we've developed two mental models that many of our customers have found useful in approaching mass migrations to the cloud.

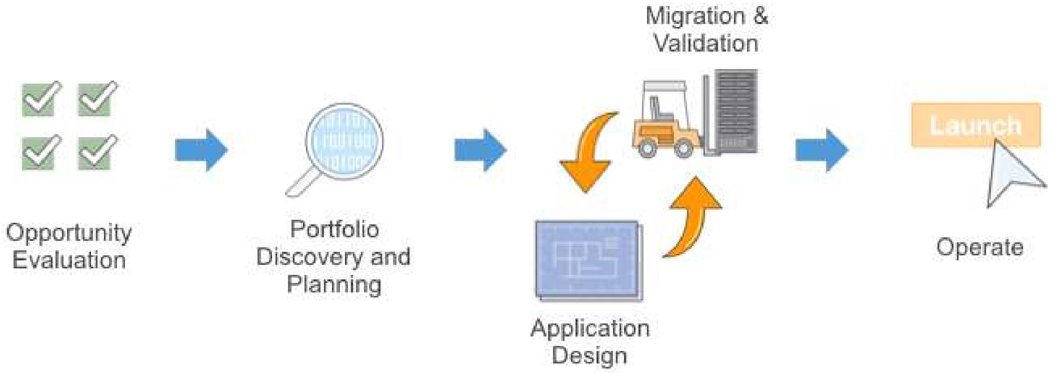
The first mental model illustrates the pattern we've seen several migrations follow. This 5-phased "Migration Process" might help you approach a migration of tens or hundreds, or even thousands, of applications.

The second mental model, which I sometimes call "The 6 R's," offers 6 different strategies for migrating individual applications to the cloud.

These mental models—while based on experience—are intended to serve as guiding principles to help you approach your migration. They are not hard-and-fast rules. Every organization has its own unique blend of constraints, budget issues, politics, culture, and market pressures that will guide its decision-making process along the way.

The Migration Process

As I mentioned above, the cloud migration process generally consists of 5 phases: Opportunity Evaluation, Portfolio Discovery and Planning, Application Design, Migration & Validation, and Operate.

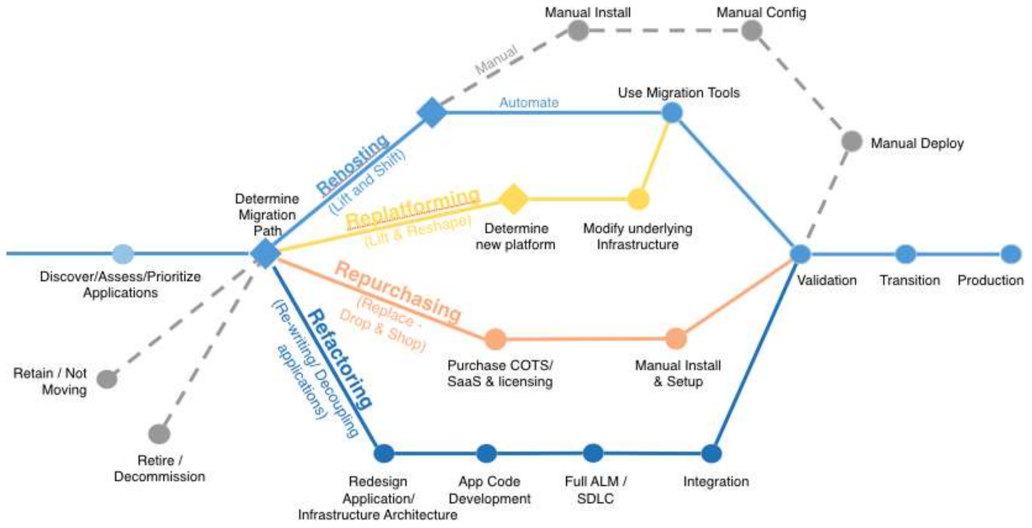


While there’s no perfect path or process for every migration, we’ve found that this mental model helps customers approach their migration, and it has allowed us (AWS) to codify the best practices, tools, and partners that organizations are using to migrate.

For an in-depth view into this “Migration Process,” see the second post in this series, A Process for Mass Migrations to the Cloud.

Application Migration Strategies: “The 6 R’s”

Applications can be migrated to the cloud in a number of different ways, and, as with the “Migration Process,” there can be many shades of gray; but we find these 6 approaches to be the most common: *Rehosting* (otherwise known as “lift-and-shift”), *Replatforming* (I sometimes call this “lift-tinker-and-shift”), *Repurchasing* (migrate to a different product/license, often SaaS), *Refactoring* (re-architect or re-imagine leveraging cloud-native capabilities), *Retire* (get rid of), and *Retain* (do nothing, usually “revisit later”). (Note: These strategies build upon the 5 R’s that Gartner outlined here in 2011.)



For a more detailed explanation of each of these 6 application migration strategies, see the third post in this series, 6 Strategies for Migrating Applications to the Cloud.

Are You Ready for a Mass Migration?

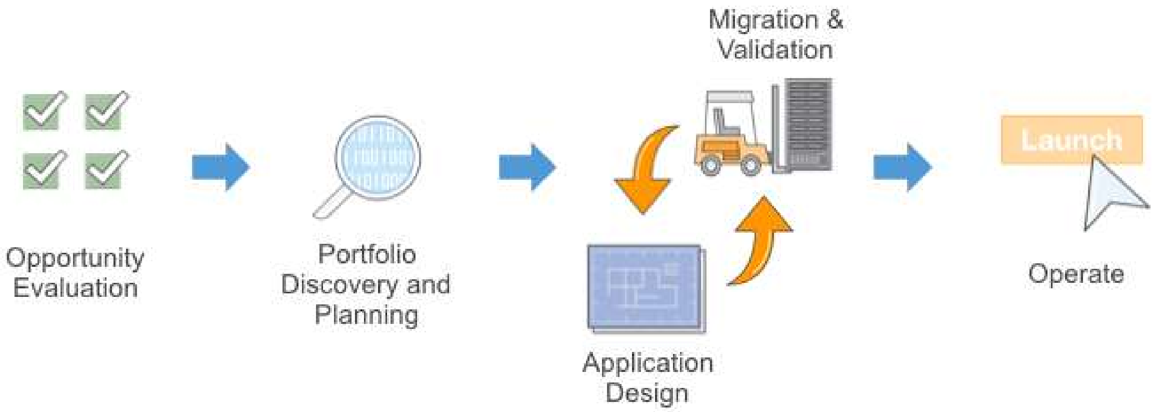
Mass migrations typically require an organization-wide effort and are most commonly embarked upon by organizations that have some experience with the cloud.

I elaborated on the activities I typically see organizations engage in before they execute mass migrations in my posts on Getting Started with the Cloud?and 4 Foundational Investments for Your Cloud Journey, although I'm finding that, as the cloud market matures and there are more success stories to draw from, more executives are contemplating mass migrations earlier in their Journey.

If you have a compelling reason to launch a migration without the benefit of existing experience, you should absolutely start to consider your business case, but I would strongly encourage you to front load the migration project with some of the activities outlined in these initial stages.

A Process for Mass Migrations to the Cloud

The "Migration Process" combines what we (AWS) know about technology migrations and some of our experience helping organizations migrate their IT portfolios to AWS. This process—while based on experience—is intended to provide some guiding principles to help you approach your migration and not meant to prescribe hard-and-fast rules. Every organization has its own unique blend of constraints, budget issues, politics, culture, and market pressures that will guide its decision-making process along the way.



Phase 1: Opportunity Evaluation

What is the business case or compelling event that will drive your migration to the cloud?

Ideally, you'll be building off some experience (see Getting Started with the Cloud and 4 Foundational Investments for Your Cloud Journey), and you'll be able to use that experience to inform your business case. In the formative stages of the cloud market, migrations were often driven by instinct—an executive who felt it was the right thing to do. As the market develops and every enterprise is considering what and how to migrate, the need for business cases and/or compelling events to drive organization-wide behavior are becoming more common.

I’m sure that I’ve yet to see every possible business case or compelling event, but I do see a lot of migrations driven by data center lease expiry, additional developer productivity, global expansion, upcoming M&A activity, and/or the drive for standardized architectures.

One customer we work with, for example, has developed a business case around developer productivity. The customer (rightfully) believes that by migrating its data centers to AWS, and training its developers in the process, each of its 2,000 developers will be 50% more productive than they are today. Driven by the elimination of wait time for infrastructure provisioning and access to more than 80 services they’d otherwise have to build/procure individually, this productivity boost will lead to an additional 1,000 years of developer capacity … each year. The customer intends to use this additional productivity to fund 100 new projects of 10 people each in an effort to find net new growth opportunities. (As a former CIO, this is probably my favorite business case yet; and, if you have a strong interest in hearing about additional business cases, please send me a note and we’ll elaborate on other cases.)

Even if your organization doesn’t require a formal business case to migrate to the cloud, I think it’s important for leaders to provide clarity of purpose and set aggressive—but achievable—goals that their organizations can rally behind. I’ve seen too many migration efforts stall without this.

As you progress in your migration, you can look to hone in on the value you’re creating, how you’re communicating that value to your organization, and become more confident in your approach to procuring IT services in a pay-as-you-go as-a-service model.

Phase 2: Portfolio Discovery and Planning

What’s in your environment, what are the interdependencies, what will you migrate first, and how will you migrate it?

This is when organizations typically inspect their configuration management databases (CMDBs), institutional knowledge, and/or deploy tools (like the AWS Discovery Service and/or RISC Networks) to deeply understand what’s in their environment. Using this knowledge, organizations can outline a plan (which should be considered subject to change as they progress through their migration and learn) on how they’ll approach migrating each of the applications in their portfolio and in what order.

The complexity of migrating existing applications varies, depending on the architecture and existing licensing arrangements. If I think about the universe of applications to migrate on a spectrum of complexity, I’d put a virtualized, service-oriented architecture on the low-complexity end of the spectrum, and a monolithic mainframe at the high-complexity end of the spectrum.

I suggest starting with something on the low-complexity end of the spectrum for the obvious reason that it will be easier to complete—which will give you some immediate positive reinforcement or “quick wins” as you learn.

The complexity will also influence how you migrate. Because it’s easy to lift-and-shift a modern application hosted on a virtualized environment, and there’s typically less technical debt associated with something developed 3 years ago versus 20 years ago, we find a strong bias toward rehosting (aka “lift-and-shift”). And, because it’s simply not possible to lift-and-shift a mainframe, we also find a strong bias toward feature rationalization and re-architecting. We (AWS and APN Migration Partners) are doing everything we can to make mainframes (and other legacy systems) easier to migrate (contact me for more details), but there’s still no silver bullet.

Phase 3 and 4: Designing, Migrating, and Validating Applications

In these 2 phases, which I often refer to together as the “migration factory,” the focus of the migration moves from the portfolio level to the individual application level, and each application is designed, migrated, and validated according to one of the 6 application migration strategies described here.

I recommend taking an approach of continuous improvement. Start with the least complex application, learn how to migrate while learning more about the target platform, and build toward the more complex application migrations as your organization becomes more cloud and migration fluent.

To help scale the “migration factory” quickly, I also recommend creating agile teams focused on some type of “migration theme.” You might have a few teams dedicated to one or more of the migration strategies, to common application types (websites, Sharepoint, back-office, etc.), to different business units, or, in all likelihood, some combination thereof. Finding themes for teams to focus on will increase the chances that they learn from common patterns and accelerate the pace at which the “factory” migrates applications. Ideally, you’ve established a Cloud Center of Excellence that can advise and guide teams on their migrations and what to expect as they progress.

Finally, make sure you have a strategy for testing and decommissioning the old systems. The good news is you shouldn’t have to purchase or wait for new hardware when you’re only going to decommission the old hardware, but you may have to run parallel environments for a period of time while you migrate traffic, users, or content. To minimize this time, make sure that each business owner is involved and ready to validate the migration in real-time, and measure the difference in cost and performance as you go. Over the next few months, I hope to have some guest posts from our APN Partners that will elaborate on the tools and patterns organizations use to do this.

Phase 5: Modern Operating Model

Finally, as applications are migrated, you iterate on your new foundation, turn off old systems, and constantly iterate toward a modern operating model.

When I was at Dow Jones, we used our migration as a forcing function to adopt a DevOps culture (more on DevOps here), and many of the executives I speak to today seek a similar path toward Agile, LEAN, or some other buzzword-friendly approach to application development.

6 Strategies for Migrating Applications to the Cloud

Formulating a Migration Strategy

Enterprises typically begin to contemplate how to migrate an application during the second phase of the “Migration Process”—Portfolio Discovery and Planning. This is when they determine what’s in their environment, what are the interdependencies, what’s going to be easy to migrate and what’s going to be hard to migrate, and how they’ll migrate each application.

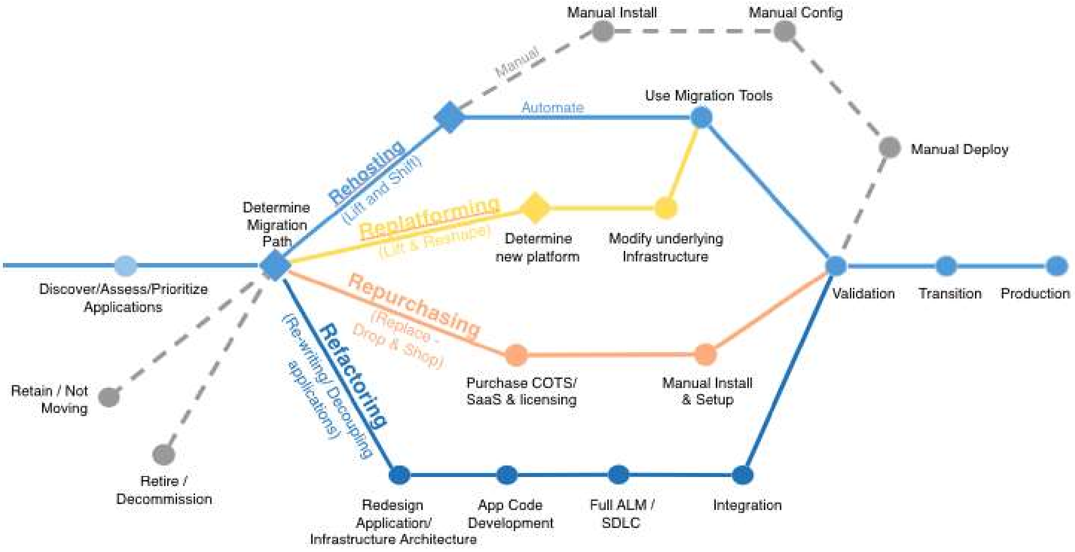
Using this knowledge, organizations can outline a plan (which should be considered subject to change as they progress through their migration and learn) on how they’ll approach migrating each of the applications in their portfolio and in what order.

The complexity of migrating existing applications varies, depending on the architecture and existing licensing arrangements. If I think about the universe of applications to migrate on a spectrum of complexity, I’d put a virtualized, service-oriented architecture on the low-complexity end of the spectrum, and a monolithic mainframe at the high-complexity end of the spectrum.

I suggest starting with something on the low-complexity end of the spectrum for the obvious reason that it will be easier to complete—which will give you some immediate positive reinforcement or “quick wins” as you learn.

6 Application Migration Strategies: “The 6 R’s”

The 6 most common application migration strategies we see are:



1. Rehosting—Otherwise known as “lift-and-shift.”

We find that many early cloud projects gravitate toward net new development using cloud-native capabilities, but in a large legacy migration scenario where the organization is looking to scale its migration quickly to meet a business case, we find that the majority of applications are rehosted. GE Oil & Gas, for instance, found that, even without implementing any cloud optimizations, it could save roughly 30 percent of its costs by rehosting.

Most rehosting can be automated with tools (e.g. AWS VM Import/Export, Racemi), although some customers prefer to do this manually as they learn how to apply their legacy systems to the new cloud platform.

We've also found that applications are easier to optimize/re-architect once they're already running in the cloud. Partly because your organization will have developed better skills to do so, and partly because the hard part—migrating the application, data, and traffic—has already been done.

2. Replatforming—I sometimes call this "lift-tinker-and-shift."

Here you might make a few cloud (or other) optimizations in order to achieve some tangible benefit, but you aren't otherwise changing the core architecture of the application. You may be looking to reduce the amount of time you spend managing database instances by migrating to a database-as-a-service platform like Amazon Relational Database Service (Amazon RDS), or migrating your application to a fully managed platform like Amazon Elastic Beanstalk.

A large media company we work with migrated hundreds of web servers it ran on-premises to AWS, and, in the process, it moved from WebLogic (a Java application container that requires an expensive license) to Apache Tomcat, an open-source equivalent. This media company saved millions in licensing costs on top of the savings and agility it gained by migrating to AWS.

3. Repurchasing—Moving to a different product.

I most commonly see repurchasing as a move to a SaaS platform. Moving a CRM to Salesforce.com, an HR system to Workday, a CMS to Drupal, and so on.

4. Refactoring / Re-architecting—Re-imagining how the application is architected and developed, typically using cloud-native features.

This is typically driven by a strong business need to add features, scale, or performance that would otherwise be difficult to achieve in the application's existing environment.

Are you looking to migrate from a monolithic architecture to a service-oriented (or server-less) architecture to boost agility or improve business continuity (I've heard stories of mainframe fan belts being ordered on e-bay)? This pattern tends to be the most expensive, but, if you have a good product-market fit, it can also be the most beneficial.

5. Retire—Get rid of.

Once you've discovered everything in your environment, you might ask each functional area who owns each application. We've found that as much as 10% (I've seen 20%) of an enterprise IT portfolio is no longer useful, and can simply be turned off. These savings can boost the business case, direct your team's scarce attention to the things that people use, and lessen the surface area you have to secure.

6. Retain—Usually this means "revisit" or do nothing (for now).

Maybe you're still riding out some depreciation, aren't ready to prioritize an application that was recently upgraded, or are otherwise not inclined to migrate some applications. You should only migrate what makes sense for the business; and, as the gravity of your portfolio changes from on-premises to the cloud, you'll probably have fewer reasons to retain.