

# DEMYSTIFYING ENTERPRISE CLOUD MODERNIZATION

## Pros & Cons Of

X-As-A-Service Offerings, Deployment Models, And Modernization Approaches

*A Solution-Agnostic Enterprise Guide For Cloud Understanding, Readiness & Migration*

W  
I  
T  
H

- Strategies For Micro & Macro Readiness Analysis—An Entire Enterprise Application Portfolio Or Single Application
- Tips for Assessing The Best Modernization Approach And Cloud Provider For Your Unique Requirements



**KEYHOLE**  
SOFTWARE

© 2020 Keyhole Software



# Table of Contents

<b>Introduction</b>	<b>3</b>
History & Progression To The Cloud	5
Cloud Computing Expenses	7
CapEx vs. OpEx	8
<b>Cloud Computing Services</b>	<b>9</b>
Infrastructure As A Service	10
Platform As A Service	12
Software As A Service	14
Function As A Service	15
<b>Deployment Models</b>	<b>17</b>
Public Clouds	17
Private Clouds	18
Hybrid Clouds	18
Multi Cloud	20
<b>Micro &amp; Macro Analysis &amp; Strategies</b>	<b>22</b>
Macro Migration Plan: Application Portfolio Level	22
Micro Migration Plan: Application Level	26
<b>Cloud Modernization Approaches</b>	<b>32</b>
Rehost	32
Replatform	34
Rearchitect	36
Repurchase	37
Retain	38
Retire	38
Potential Improvement Points	39
What Is The Best Approach?	43
<b>Assessing Cloud Providers</b>	<b>45</b>
<b>Conclusion</b>	<b>51</b>



## KEYHOLE SOFTWARE

Keyhole Software is a software development and consulting firm headquartered in Kansas City, with active teams located nationwide.

Our **expert employee consultants** excel as development “change agents,” helping our clients to be successful with software technologies and approaches like the cloud.

We frequently assist clients through custom application design, development, and modernization initiatives with Java, JavaScript, and .NET technologies.



Standard  
Consulting  
Partner

## CLOUD SERVICES

We assist clients at every step in the cloud adoption process including:

### Cloud Readiness Assessment

Assessing current status and strategically planning the best path for successful cloud adoption, including agnostic vendor analysis and migration strategy for bottom-line savings.

### Application Cloud Migration

Migration of existing applications to PaaS or cloud offerings, including lift-and-shift, rearchitecting, replatforming, and others.

### Cloud Solution Development

Quickly creating and deploying mission critical web apps and POCs that scale with your business.

### Cloud Education

Teaching your team to be successful with DevOps, orchestration, and cloud techniques required in a successful implementation.

“

Helping our clients leverage cloud technologies to deliver applications faster, with higher levels of quality, scalability, and innovation.

”



# Introduction

This white paper includes a solution-agnostic resource to understand enterprise cloud modernization and migration. Key topics of the paper include:

- History and progression of cloud technology and what it means for business financials
- An introduction to computing services Platform as a Service (PaaS), Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Function as a Service (FaaS) with key advantages and disadvantages of each option
- A look into deployment models your organization can consider including public, private, hybrid, and multi cloud—each with associated pros and cons
- Strategies for micro & macro cloud readiness analysis and a path for doing so
- The 6 R's of Cloud Migration with sample use cases and advantages and disadvantages for each one
- Tips for assessing a cloud provider with a general overview of the top three

Whether you are a software development company or an organization located in a different vertical like healthcare or finance, you have software to manage your operational business. This generally includes external-facing applications or internal business applications that connect to line-of-business applications that form the bedrock of the company—mainframe systems that run factories or fulfillment centers, HR, payroll, SAP, CRM, and the like.

If you are a large organization, those mission-critical applications are likely located on a complex infrastructure that was grown over decades. That infrastructure generally must support multiple platforms, with many different types of data as well as highly customized legacy applications. In many cases, those companies already have prior application modernization investments which now appear outdated, inflexible or are simply ineffective because the technology has changed since they were built.

Many IT budgets are consumed by maintaining those existing applications and resources. Getting out from under the weight of these existing applications can be daunting—schedule is king, and making time to address modernization comes at the expense of current business needs.

As the business world moves into the digital space to prepare to meet the next generation of customers to compete globally, organizations can't manage their applications in the same way as they have in the past—they must take them to the next level.



## Common Business Needs

In any modern IT department, there is generally a consensus of common business needs that transcend industry verticals. These goals can include:

- Delivering the right applications for the business faster and with higher levels of quality, scalability, and capacity
- Minimizing expenses directed toward outdated, inefficient applications and redirecting budget to innovation
- Finding a flexible way to connect legacy applications with new to proactively support new business opportunities
- To better understand the current state of their organization's infrastructure, code, data, DevOps, debt, and team maturity and how each piece affects future innovation

Cloud technologies provide many of the building blocks to help businesses achieve these goals and prepare themselves for the future. Are your applications ready for this new tomorrow?

## History & Progression To The Cloud

The concept of Computing As a Service has been around as far back as the 1960s. Back then, rather than having to buy a dedicated mainframe themselves, computer bureaus would allow companies to rent time on a mainframe.

So in the beginning, there were physical machines. They were expensive, and developers loaded them up with multiple applications to save costs. This caused no end of conflicts and unexpected bugs.

The next iteration was virtual machines. In the 1990s, telecommunications companies began virtual private network (VPN) services with comparable quality of service, but at a lower cost. By switching traffic as they saw fit to balance server use, they could use overall network bandwidth more effectively. Scientists and technologists later explored ways to make large-scale computing power available to more users through time-sharing.<sup>1</sup>

Since multiple VMs could be run on a single physical box, developers installed their applications in contained VMs, which reduced conflicts and improved utilization. The downside of this was dealing with shared memory and unpredictable performance.

---

<sup>1</sup>Griffin, Ry'mone. [Internet Governance](#). Scientific e-Resources. Nov 20 2018 ISBN 978-1-83947-395-1.



Enter the cloud. Amazon introduced its Elastic Compute Cloud (EC2) In August 2006,<sup>2</sup> Microsoft announced Microsoft Azure in October 2008 (released February 2010),<sup>3</sup> and Rackspace Hosting and NASA jointly launched an open-source cloud software initiative In July 2010.<sup>4</sup> Companies of all sizes could now rent the virtual machines they needed, when they needed them. They could scale applications up and down on demand.

Today, more organizations are evaluating PaaS (Platform as a Service) and serverless approaches to application development, moving beyond cloud-hosted VMs, removing the need to manage servers and scalability challenges and increasing the speed of the iterative development process.

As the technology has improved and more solutions providers have joined the fold, the number of offerings available to engage with have increased significantly.

## Definition of Cloud Computing

The National Institute of Standards and Technology's definition of cloud computing identifies "five essential characteristics:"<sup>5</sup>

1. **On-Demand Self-Service:** A consumer can unilaterally provision computing capabilities, such as server time and network storage, automatically as needed without requiring human interaction with each service provider.
2. **Broad Network Access:** Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).
3. **Resource Pooling:** The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model with different physical and virtual resources dynamically assigned and reassigned according to consumer demand.
4. **Rapid Elasticity:** Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear unlimited and can be appropriated in any quantity at any time.

<sup>2</sup> "[Announcing Amazon Elastic Compute Cloud \(Amazon EC2\) – beta](#)". AWS. Accessed 2 April 2020.

<sup>3</sup> "[Windows Azure General Availability](#)". The Official Microsoft Blog. Microsoft. 2010-02-01. Accessed 28 March 2020.

<sup>4</sup> [OpenStack](#). Wikipedia. Accessed 2 April 2020.

<sup>5</sup> [The NIST Definition of Cloud Computing \(Technical report\)](#). National Institute of Standards and Technology: U.S. Department of Commerce. Peter Mell; Timothy Grance (September 2011).



5. **Measured Service:** Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled and reported, providing transparency for both the provider and consumer of the utilized service. — National Institute of Standards and Technology<sup>6</sup>

## Cloud Computing Expenses

Establishing and running a physical, on-premise data center is expensive. The traditional way to plan for unexpected growth has always been to purchase and keep additional servers, storage, and licenses in reserve. In our experience, many enterprise companies spend the majority of their IT budgets on simply “keeping the lights on”—typically 80 percent or more spent on operating, maintaining and managing existing systems and workloads.<sup>7</sup>

Comparatively, the cloud offers dynamic, “on-demand” provisioning of resources in near real-time. This gives the ability to scale up when the usage need increases and down if resources are not being used.<sup>8</sup> Most cloud computing services use a subscription-based model where you pay only for the resources actually consumed. This allows customizability so your company can choose a package that suits your budget.

Across industries, cloud continues to be one of the fastest-growing segments of IT spend. Through 2022, growth in enterprise IT spending for cloud-based offerings will be faster than growth in traditional (non-cloud) IT offerings<sup>9</sup>, making cloud computing one of the most disruptive forces in IT markets since the early days of the digital age.

For example, Gartner’s latest IT spending forecast shows that spending on data center systems was forecasted to be \$195 billion in 2019, but down to \$190 billion through 2022. In contrast, spending on cloud system infrastructure services (IaaS) will grow from \$39.5 billion in 2019 to \$63 billion through 2021.<sup>10</sup>

---

<sup>6</sup> [The NIST Definition of Cloud Computing \(Technical report\)](#). National Institute of Standards and Technology: U.S. Department of Commerce. Peter Mell; Timothy Grance (September 2011).

<sup>7</sup> [The Legacy IT Conundrum: Money Pit or Value-Add?](#) CIO. 5 March 2014.

<sup>8</sup> Mao, Ming; M. Humphrey (2012). A Performance Study on the VM Startup Time in the Cloud. Proceedings of 2012 IEEE 5th International Conference on Cloud Computing (Cloud2012).

<sup>9</sup> [Cloud Shift Impacts All IT Markets](#). Gartner. Accessed 1 April 2020.

<sup>10</sup> [Cloud Shift Impacts All IT Markets](#). Gartner. Accessed 1 April 2020.



That isn't to say that cost savings is a guarantee. As a caveat, don't assume you will automatically save money by moving to the cloud unless you have done the work of honestly analyzing your cloud needs, as detailed in further sections.

## CapEx vs. OpEx

From a manager's perspective, the main difference found in adopting cloud computing removes the need to purchase large infrastructure hardware to run your applications. Instead, you will now have a monthly cost that directly relates to your usage of the cloud.

But, from an accountant's perspective, migrating to the cloud has significant implications in the financial statements - specifically moving from Capital Expenses to Operating Expenses. CapEx has been the standard model of traditional IT procurement, while OpEx is how cloud computing services are procured. The two have very different implications for cost, control, and operational flexibility.

### CapEx

With CapEx accounting procedures, the money spent on fixed assets—like the purchase of physical property or equipment used by employees—is capitalized on the balance sheet rather than directly expensed to the income statement.

These purchases are significant, generally lump sum and upfront investments to acquire and maintain your critical assets. Companies often finance the purchase of long-term assets. A CapEx asset's useful life most likely extends beyond a year. Therefore, the value is depreciated, allowing for tax deductions over time as the purchase accumulates wear and tear.

From an IT perspective, CapEx can include capital expenditures like: investing in new network equipment to improve performance, building a server farm to host home-grown or business applications, or real estate property where your IT staff conducts business.

### OpEx

Conversely, with OpEx, think of the operating costs required to run your business. This is a monthly or annual recurring investment.

Cloud costs are based on consumption which generally means that there is less risk of overprovisioning. Line-of-business units can also take ownership of budgets that were once exclusively of the IT department. It is accounted for tax-wise in the current tax year (no depreciation).





From an IT perspective, OpEx can include operating costs like: Public cloud consumption, offsite backup/archival storage, and contract items such as maintenance agreements, website hosting, and web domain registrations.

CapEx	OpEx
<b>Purpose</b> Assets purchased with useful life beyond current tax year	<b>Purpose</b> Ongoing costs to run a business
<b>When Paid</b> Up front in a lump sum	<b>When Paid</b> Reoccurring, monthly or annual
<b>Listed As</b> Property or equipment	<b>Listed As</b> Operating Cost
<b>Tax Implications</b> Deducted over time as asset depreciates	<b>Tax Implications</b> Deducted in current tax year
<b>Example</b> Purchase of servers and networking equipment	<b>Example</b> Public cloud consumption and services subscription

Figure 1: Capital Expenditures vs. Operational Expenditures

Without the need to leverage debt or account for the purchase as a capital expenditure, the process of approval for cloud management should require fewer checkpoints and be easier in general. The expenses will directly relate to the usage and costs of the cloud, rather than the depreciation and overhead costs of the facility and equipment.

## Cloud Computing Services

There are different types of cloud platforms available on the market for your business to consider. To select the right one, you must understand the level of control you desire, the pros and cons of each approach, and how they could impact your applications.

Let's introduce an analogy. Consider a cloud platform as the home for all your applications. For some applications, you want full control over every square foot—infrastructure, data, and code. For these applications, **Infrastructure as a Service (IaaS)** is the way to go.

*IaaS is like building your own custom home for applications – you can build it to your unique specifications using an experienced builder. You have complete control, but their crews do the actual groundwork and heavy lifting. Once you move in, you must perform maintenance and upkeep.*

For other applications, you just want to move them in and run them as efficiently as possible, without thinking about plumbing and wiring. This is great for **Platform as a Service (PaaS)**.



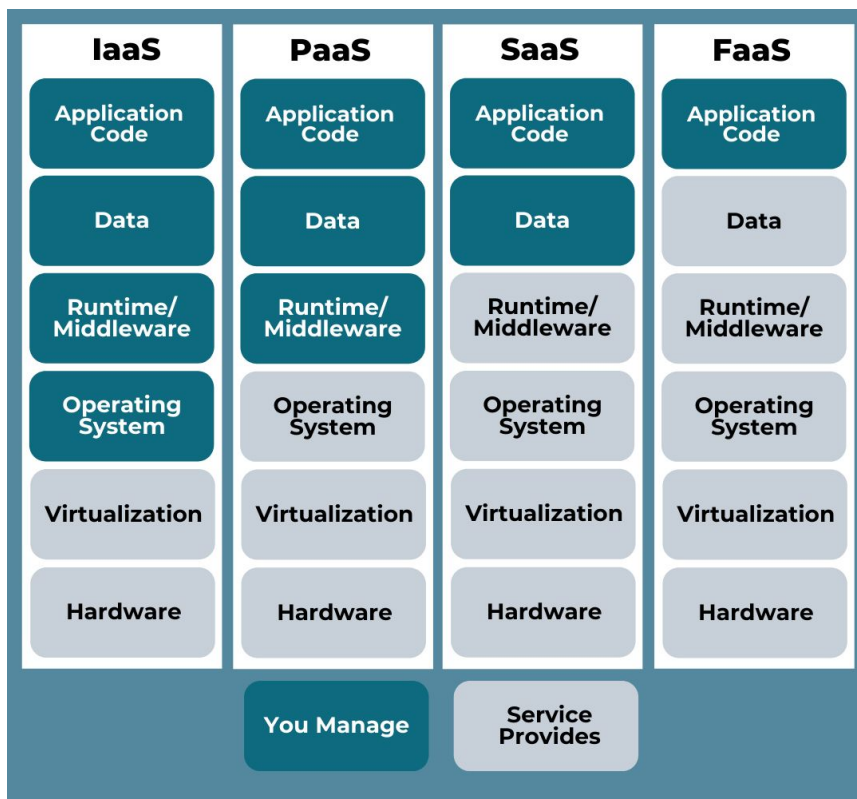
*PaaS is like a bed and breakfast for applications. You move in, and things are handled for you. Your bed is kept made and surfaces are dusted; you can focus on your daily activities.*

For other applications, you simply don't have time for any details and want it handled for you. **Software as a Service (SaaS)** is available as-is and out-of-the-box. Every aspect (infrastructure, platform, and application) is managed for you.

*SaaS applications are more like staying at a luxury hotel. Once you check in, your concierge will handle your every want and need of available amenities.*

With **Function as a Service (FaaS)** or "serverless" you can choose which platform to run, what language to write. The infrastructure is managed by your cloud provider. You just use a particular service when you need it.

*FaaS or "serverless" is like renting a ballroom at a luxury hotel for a corporate event. You reserve the space for a set event, the venue staff takes care of your guests, catering, and decorations.*



One piece of great news is that it is not an either-or choice. You can own a home and stay in a hotel or B&B at the same time. As such, you can run some of your cloud applications in IaaS, some in PaaS, and some using SaaS—it's all up to you.

Figure 2: Comparison of IaaS, PaaS, SaaS, and FaaS



# Infrastructure As A Service

IaaS cloud infrastructure most closely resembles having your own data center. It offers the greatest level of control and power over software and hardware. This is the lowest level a cloud provider can offer, only supplying the bare infrastructure and you as a customer are in charge of putting the pieces together.

In the context of IT, infrastructure has three primary components: Network, Storage and Computing. In the past, these three components were most often managed internally and on-premise by an organization. Responsibilities such as system backups, redundancy, load balancing, housing servers, and network availability were in the hands of the organization.

IaaS moves all of these responsibilities to the cloud provider and gives the subscriber access to VMs (Virtual Servers) through the cloud platform. A basic IaaS offering provides VMs with either specific, fixed, or dynamic VM-sizing options. Cloud providers might offer multiple VM sizes at fixed prices per hour, day, week, or month.

With IaaS, the cloud provider will normally install the chosen operating system, but the subscriber is still responsible for setting up and maintaining the system software. They must have the knowledge of a mature operations model and rigorous security stacks. The subscriber must also keep the system current with OS and application updates.

## IaaS Advantages

- **Cost Savings:** Expensive hardware and network infrastructures are managed by the cloud provider, which drastically reduces expenses. Additionally, you generally only pay for what is actually used.
- **Freedom and Control:** An IaaS provider is responsible for the entire infrastructure, but subscribers have total control over it. You still have control over how your server (VM) is set up and configured. Overall there is higher flexibility and customization.
- **On-Demand Scalability:** If you need additional VMs, you can scale out and add additional servers on the fly and scale back when extra power is no longer necessary. If you need to scale up, you can add additional CPUs, memory, and disk space on-demand as well.
- **High Reliability and Redundancy:** Cloud providers make huge investments in creating infrastructures that are both extremely performant and can handle outages with different failover strategies. Redundancy can be regionally and geographically distributed. They also offer options to handle disaster recovery scenarios



- **Less Vendor Lock In:** Out of all of the “X-as-a-Service” options, IaaS is the least likely of the delivery models to result in vendor lock in.<sup>11</sup>

## IaaS Disadvantages

- **More Extensive Technical Knowledge Required:** Out of all of the “X-as-a-Service, Platforms, this requires the most technical knowledge. With more freedom comes more responsibility. To have success with this level of control, you must have personnel that are trained for infrastructure best practices and can manually configure everything in the cloud stack from the operating system layer and above.
- **Subscriber Responsible For OS-Level Configuration:** Subscriber is responsible for handling all server-side configuration and maintenance including load balancing, operating system updates, installing and maintaining apps and operating systems, as well as for runtime, middleware and data.
- **Security:** In addition to managing the configuration and maintenance of the VM, you’re also responsible for making sure that the server is locked down, secured, and doesn’t have any security vulnerabilities. Some IaaS providers manage this.

## Select IaaS Cloud Computing Service Providers

- AWS Elastic Compute Cloud (EC2)
- Microsoft Azure
- Google Compute Engine
- DigitalOcean
- Apache CloudStack

## Platform As A Service

In a PaaS environment, a cloud supplier provides only a specific platform that users can build applications on top of. Like IaaS, the cloud provider still provides the VM with an installed operating system. However, instead of simply giving you full access to the VM instance to manage yourself, you are given access to a complete development and deployment environment.

Unlike the previously discussed IaaS, with PaaS, the cloud provider handles all of the operating system maintenance and updates as well as CPU types, memory, RAM, storage, networking etc. It’s designed to be easier to use with less configuration when compared with IaaS.

---

<sup>11</sup> [SaaS vs. PaaS vs. IaaS – Where the Market is Going](#). Business 2 Community. Published May 28, 2019. Accessed 9 April 2020.



As a comparison between the previous IaaS and PaaS, let us suppose you need to run an Angular application that utilizes a .Net Web API REST service:

- With IaaS, you would need to perform all necessary server-side setup and configuration for hosting the applications.
- With PaaS, you would simply deploy the applications to a service container and all the low-level configuration is handled by the cloud platform.

Like IaaS, PaaS does include infrastructure—servers, storage, and networking. But PaaS also includes middleware, development tools, business intelligence (BI) services, database management systems and more. PaaS is designed to support the complete web application lifecycle: building, testing, deploying, managing and updating rather than just its infrastructure.

## PaaS Advantages

- **Cost Savings:** The development and deployment environment is provided by the cloud provider which drastically reduces expenses. Additionally, you generally only pay for what is actually used.
- **Decreases Overhead:** Completely eliminates the need for the maintenance of in-house hardware infrastructure. PaaS removes the responsibility of managing a VM at the OS level and gives you control at the application level through various development and deployment options.
- **On-Demand Scalability:** PaaS offers the same scalability benefits IaaS offers. You can add additional servers on the fly and scale back in when extra power is no longer necessary. If you need to scale up, you can add additional CPUs, memory, disk space, on-demand as well.
- **Reliability and Redundancy:** PaaS offers the same reliability/redundancy benefits that IaaS offers. Cloud providers make huge investments in creating infrastructures that are both extremely performant and can handle outages by utilizing different failover strategies. They also offer options to handle disaster recovery scenarios.
- **Improved Time to Market:** With PaaS, developers have access to numerous automated tools and technological innovations which can significantly speed up the creation of apps.

## PaaS Disadvantages

- **Some Vendor Lock-In:** Each PaaS offering tends to be specific to each vendor, which can limit your options. It can be difficult to migrate to another option or to stop using a service when it is baked-in to your operations.



- **Technical Knowledge Required:** To have success with PaaS, your personnel must understand how to customize the basic application stack setup provided.
- **Loss of Control at the Operating System Level:** With PaaS, you no longer have control at the operating system level, and you cannot manage things like when patches and updates are applied to the OS.
- **Security:** With PaaS, the responsibility of security is shared between you and the provider. As the customer, you are responsible for securing data, applications, and user access, whereas the provider secures the physical infrastructure and operating systems.

## Select PaaS Cloud Computing Service Providers

- AWS Elastic Beanstalk
- Heroku
- OpenShift
- Google App Engine
- Pivotal Cloud Foundry
- Amazon Web Services (AWS)
- Oracle Cloud Platform (OCP)
- Microsoft Azure
- IBM Cloud platform
- SAP Cloud Platform
- Engine Yard

## Software As A Service

SaaS is a complete application delivered as a service via a third-party over the internet to the consumer. You purchase SaaS on a subscription basis from a cloud provider. It's common for an organization to pay a subscription fee (generally billed monthly or yearly) for the use of an application. Your users connect to your application online via a web browser, which also means that it is OS-agnostic.

All of the underlying infrastructure, middleware, O/S, and application data are located in the cloud provider's data center. The cloud service provider manages the hardware and software to ensure the availability and security of the application and data with the appropriate service agreement.

## SaaS Advantages

- **Easiest To Use:** You don't need to purchase, install, update, or maintain any hardware, middleware, or software. SaaS makes even sophisticated enterprise applications, such as ERP and CRM, affordable for organizations that lack the resources to buy, deploy, and manage the required infrastructure and software themselves.
- **Reduced Costs:** With SaaS, you generally pay only a fixed monthly subscription based upon the number of users needing access to the hosted application. It is typically cheaper over time when factoring in associated equipment and staffing costs of running an internal system.



- **Scalability:** As your need to add more users increases, you simply purchase more subscriptions.
- **Accessibility:** SaaS offerings are typically browser or mobile-based and don't require any special setup on a user's device.

## SaaS Disadvantages

- **Vendor Lock In:** Of all of the “X-as-a-service” offerings, SaaS is the most restrictive when it comes to lock-in. You may be able to take your data, but it's the SaaS provider's application.
- **No Control Of Security:** It's up to the software provider to implement security best practices, and in most cases, you have no control over how that is administered.
- **Lacks Ability to Customize Software:** Most SaaS solutions don't offer the flexibility to customize an application to meet needs that are specific to your organization.
- **Lack of Control:** You fully depend on the vendor and have zero control over the cloud-based infrastructure it runs on. If the software provider experiences an outage, so do you.

## Select SaaS Service Providers

- Google G Suite / Google Apps
- Salesforce
- Slack
- Microsoft Office 365
- Dropbox
- DocuSign
- HubSpot
- Zendesk
- Canva
- Mailchimp

## Function As A Service

Instead of deploying an entire application to one or more servers, with FaaS you install a function—a piece of business logic or part of an application. The functions are only loaded when needed and can be executed on demand.

As the name suggests, it's based on the functions which can be triggered by a given event. Think of it as an event-based architecture where the trigger can be event types like file/object updates, scheduled tasks, or messages added to a message bus.

The level of simplicity is so high that it's often called a serverless architecture. Of course, servers are still used to execute code, but an additional layer of abstraction is added to remove the developer from any sort of server configuration. The developer just writes a function and doesn't have to be concerned with deployment, server



resources, or scalability as the cloud provider runs the server and dynamically manages the allocation of machine resources.<sup>12</sup>

You will notice that FaaS is similar to PaaS in that it also hides "servers" from developers. However, such hosting services typically always have at least one server process running that receives external requests. By contrast, FaaS does not require any server process to be run.

FaaS isn't always the best option, or even possible, for some applications. There are design constraints—each invocation must be stateless. RESTful and other event-driven applications are a good fit, and so is work that runs on a schedule. Use cases are generally associated with "on-demand" functionality that enables the supporting infrastructure to be fully powered down when not in use. Example: data processing like extract-transform-load (ETL), batch processing, and IoT.

## FaaS Advantages

- **Lower Costs:** Developers only pay for function execution time (not process idle time).
- **Productivity Increase:** Units of code exposed to the outside world are simple, event-driven functions, which simplify the task of back-end software development.
- **Elastic Rather than Scalable:** It inherently scales 100% down as well as up and that scale is managed by the cloud provider.

## FaaS Disadvantages

- **Specific Use Cases:** Only a good fit for certain types of applications that don't maintain state (or rely solely on external resources for it).
- **Performance and Latency:** An initial cold start request may take longer to be handled than an application hosting platform where a server is already engaged.
- **Execution Duration:** FaaS functions are typically limited in how long each invocation is allowed to run.
- **Monitoring & Debugging:** Diagnosing excessive resource usage problems with serverless code may be more difficult as there is typically no ability to dig into more detail or replicate an issue in a local environment.
- **Security:** The total attack surface can be significantly larger when compared to traditional architectures. There are many more components to the application and each component is an entry point to the serverless application.

---

<sup>12</sup> [Serverless Architectures](#). Fowler, Martin. Accessed 31 March 2020.





## Select FaaS Cloud Computing Service Providers

- AWS Lambda
- Google Cloud Functions
- Microsoft Azure Functions
- IBM/Apache's OpenWhisk
- Oracle Cloud Fn

## Deployment Models

You'll find there are different ways to deploy cloud resources. Options for deployment include public, private, and hybrid clouds. All three scenarios provide benefits including cost-effectiveness, performance, reliability, and scale. However, which deployment method you choose must depend on your business needs.

### Public Clouds

This approach is the most common way of deploying cloud computing. The cloud resources (like servers and storage) are owned and operated by a third-party cloud service provider and delivered over the Internet.

Public clouds can include SaaS, PaaS, and IaaS services. Examples of public cloud providers include AWS, Azure, and Google Cloud Platform.

#### Public Cloud Advantages

- **Cost Savings:** You pay only for the service you use, and you are essentially outsourcing costs to a third party who can handle them more efficiently.
- **No Maintenance:** Your service provider provides the maintenance, so you don't have to.
- **Near-Unlimited Scalability:** On-demand resources are available.
- **High Reliability:** A vast network of servers protects against failure.
- **Higher Security (For Some):** Many small or medium businesses may not have the resources to implement strong security measures. This

outsources some aspects of cybersecurity to a larger provider with more resources.

#### Public Cloud Disadvantages

- **Weaker Security (For Some):** Weaker security sometimes is viewed as the main disadvantage in public cloud service. Most of the clouds have excellent security measures in place. However, trust in any third party is often an issue when confidential information is managed.
- **Compliance Concerns:** Multitenancy is when multiple customers of a cloud provider are accessing the same server. Multitenancy might be a concern for businesses that need to meet strict regulatory compliance standards. Note: certain cloud providers allow for the option to host on dedicated or isolated servers to



eliminate multi-tenancy concerns (Azure and AWS, for example).

- **Vendor Lock-In:** This is always a concern with cloud technology; the

business can end up reliant upon the external cloud vendor's services to maintain its internal business operations.

## Private Clouds

A private cloud is a cloud infrastructure operated solely for a single organization, whether managed internally or by a third party. It is maintained on a private network and physically located either on that organization's site or hosted by a third-party service provider. A private cloud can make it easier for an organization to customize resources to meet specific IT requirements.

To create your own private cloud, you can use a platform like OpenStack or VMware's vCloud.

Private clouds are often used by government agencies, financial institutions and any other mid to large-sized organizations seeking enhanced control over their environment.

### Private Cloud Advantages

- **Enhanced Flexibility:** Your organization can customize its cloud environment to meet specific business needs.
- **Improved Security:** Resources are not shared with others, so higher levels of control and security are possible.
- **High Scalability:** Private clouds still afford the scalability and efficiency of a public cloud.

maintenance, private clouds are generally more expensive than public clouds.

- **Maintenance:** Set up, support, and maintenance for private clouds is more time-consuming. A private cloud service requires in-house IT administration.

### Private Cloud

#### Disadvantages

- **Higher Cost:** Because they require both hardware and

## Hybrid Clouds

A Hybrid Cloud can often provide the best of both worlds between public and private. Gartner defines a hybrid cloud service as a cloud computing service that is



composed of some combination of private, public and community cloud services, from different service providers.<sup>13</sup>

In this scenario, your resources are spread over both private and public platforms with connections that you monitor. You may combine on-premise infrastructure, private clouds, and public clouds to reap the most helpful advantages. In a hybrid cloud, data and applications can move between private and public clouds for greater flexibility and more deployment options.

Additionally, IT organizations can use public cloud computing resources to meet temporary capacity needs that cannot be met by the private cloud. This is known as “cloud bursting,” an application deployment model in which an application runs in a private cloud or data center and “bursts” to a public cloud when the demand for computing capacity increases.<sup>14</sup> The benefit lies in that you scale and pay for extra compute resources only when they are needed.

## Usage Scenarios

You might, for example, move application code and infrastructure to the cloud but decide to keep your data on-premise for security and governance reasons. This is a common hybrid scenario that addresses specific requirements in regulated industries like finance or healthcare.

As another example, imagine that your web application is quickly gaining users and popularity. In order to keep up with the increasing demand, you need the underlying resources to scale dynamically. Your goal is that during peak usage, the maximum resources are deployed to serve requests, but when demand drops, unneeded resources are simply dropped to save money. This scenario is certainly available with a *public cloud*, but suppose the data your application gathers is highly confidential and cannot be stored off-premise. This is where a hybrid solution can help. In this case, you can choose which components you want to “live” in the public cloud and which will remain in your data center.

## Hybrid Cloud Advantages

- **Reduced Vendor Lock-in:** Less risk of being unable to switch providers.
- **Control:** Maintain a private infrastructure for sensitive assets.
- **Backups to Avoid Downtime:** If one cloud crashes or breaks, a company can rely on the other cloud, avoiding service interruptions.
- **Flexibility:** Take advantage of additional resources in the public cloud only when you need them.

---

<sup>13</sup> "[Mind the Gap: Here Comes Hybrid Cloud – Thomas Bittman](#)". Thomas Bittman. Retrieved 26 March 2020.

<sup>14</sup> [What is cloud bursting?](#) Microsoft Azure. Accessed 19 May 2020.



- **Cost-Effectiveness:** The ability to scale to the public cloud, so you pay for extra computing power only when needed.
- **Great For Data Redundancy:** Data and backups can be stored in more than one location for increased stability.
- **Gradual Migration:** Transitioning to the cloud doesn't have to be overwhelming because you can migrate gradually by phasing in workloads over time.
- **Cloud Bursting:** Cloud bursting is an application deployment model in which an application runs in a private cloud or data center and "bursts" to a public cloud when the demand for computing capacity increases. A primary advantage of cloud bursting and a hybrid cloud model is that an organization pays for extra compute resources only when they are needed.

## Hybrid Cloud Disadvantages

- **Greater Attack Surface:** Whenever network infrastructure becomes more complex, there is a greater chance that an attacker will find a vulnerability to exploit.
- **More Complex Integrations:** The connection and orchestration processes have more steps to setting up a hybrid cloud compared to deploying a single public cloud or a single private cloud, as any connecting technology (such as a VPN) must be established and maintained too.

## Multi Cloud

Multi Cloud refers to the distribution of software, applications and assets across more than one cloud hosting environment and provider. In a recent Gartner survey of public cloud users, 81% of respondents said they are working with two or more providers.<sup>15</sup>

Multi Cloud and Hybrid Cloud approaches are not mutually exclusive: you can have both, simultaneously. For example, an enterprise may concurrently use separate cloud providers for infrastructure (IaaS), platform (PaaS) and software (SaaS) services.

Multicloud can also take a functionality-based approach. It might use Amazon EC2 for your compute power and Redshift for the data warehouse while using IBM Bluemix's Watson as the artificial intelligence platform. Or an organization may use Amazon Web Services as its primary cloud infrastructure provider, but decide to use Google for big data and analytics.

It could also use different infrastructure providers for different workloads, either equally balanced across multiple providers, or with one active provider and the other

---

<sup>15</sup> ["Why Organizations Choose a Multicloud Strategy"](#) - Smarter With Gartner. Retrieved 1 April 2020.



as a backup. For example, you could run your application primarily on Azure but is completely replicated and backed up on AWS should there be a service interruption.

In addition to the significant benefit of fault tolerance, one reason this can occur is the desire to minimize vendor lock-in amid the dominance of mega-vendors in the public cloud services market.

With more providers certainly comes with additional complexity. Automation is key to ensure low-level and monitoring tasks don't require excess time. It is important to focus on policy standardisation and automation of those policies to automatically apply to each cloud environment.

## Multi Cloud Advantages

- **Best Platforms For Specific Functions:** Ability to pick and choose the best-of-breed services from every cloud platform.
- **Enhanced Resilience:** In the case of an outage or disruption with one provider, the impact can be mitigated automatically by relying on the other provider already established.
- **Significant Reduction In Vendor Lock-in:** Less risk of being unable to switch providers.
- **Cloud Cost Optimization:** You can identify the options that will meet your workload performance requirements at the cheapest cost.
- **Low Latency:** Users are widely distributed geographically, multiple providers can address latency issues. For example, deploy data centers to multiple regions according to user locations. The datacenter closest to end-users then serves the requested data with minimum server hops.

## Multi Cloud Disadvantages

- **Enhanced Complexity, Application Sprawl:** More data in more places means more systems to keep online at any given time. More providers means multiple billing terms and statements. Toggling between cloud providers to perform tasks can be complicated. Greater care must be taken to ensure the whole cloud picture is cohesive with the strategy.
- **Greater Attack Surface:** Whenever network infrastructure becomes more complex, there is a greater chance that an attacker will find a vulnerability to exploit.
- **More Complex Integrations:** The connection and orchestration processes have more steps to setting up a hybrid cloud compared to deploying a single public cloud or a single private cloud, as any connecting technology (such as a VPN) must be established and maintained too.





# Micro & Macro Analysis & Strategies

## Macro Migration Plan: Application Portfolio Level

Enterprise-level organizations are generally owners of more than one application; sometimes IT governs an entire suite of disparate applications.

While it is important to consider the best path for each individual application, it's also important to have a comprehensive and cohesive approach for all applications in your organization. The goal is that when you move to the cloud, you have a strategic plan for your entire portfolio landscape to best actualize cloud benefits through the lowest investment and labor required.

Perhaps the most important part of this process of understanding the current set of applications, data assets and their relationship to business functions. The goal is to determine where the current suite of applications fits in the long-term IT strategy and how well it actually supports evolving business models.

If approached correctly, portfolio modernization is not a big-bang effort that replaces the entire suite of applications with new software. Portfolio modernization is a strategic effort that will identify and prioritize applications based on their business impact.

Migrating to the cloud is a transformation that requires broad organizational change. As with any large-scale change, it doesn't work when all people aren't on the same page. Reach out to key people throughout the organization—both from the IT side and business owners. Getting all engaged and supported before you migrate will generally lead to a smoother, faster cloud migration process.

### Assessment of Application Portfolio

An Application Portfolio Assessment helps an organization better understand the entirety of its existing application environment. The key focus of the assessment is an understanding of an application's business value and technical characteristics through Application Rationalization. These two dimensions will help define both the modernization technique to be used and the potential value to the business associated with the change.



To go through the process to determine your migration plan, you must analyze each of your applications to best determine unique constraints and opportunities. You will need to know the answers to questions like the following:

- Which applications are the most important to the business?
- Which applications will be the easiest to move?
- Which application moves would have the biggest impact?
- What are the business priorities of your stakeholders?
- Are there commercial off-the-shelf alternatives that could accomplish the same thing at a lower cost?
- What skills will be necessary to move each application—do you have them in-house already?

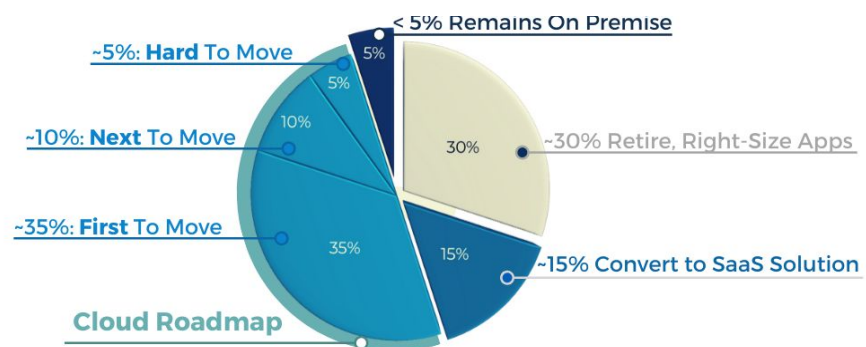
Strategies for best assessing at the application level are in the [micro-strategy section](#) that follows.

## Application Portfolio Rationalization

Fortune 500 IT organizations that are looking to modernize a sprawling application portfolio with thousands of applications of varying size and complexity can expect to fall into a similar distribution of application status after the rationalization process.

*Figure 3: Large-Scale Application Rationalization*

Most organizations discover that cloud modernization provides a great opportunity to ‘clean house’ by retiring or re-sizing applications



that are no longer used or no longer valuable. Microsoft states that usually about 30%<sup>16</sup> of a large enterprise’s application portfolio could simply be retired. Bonus: this generates immediate savings.

A big chunk of this group to be retired includes:

- The dreaded shadow IT—departmental applications that run under a desk somewhere and are outside of central control.
- Applications that were implemented to meet a tactical need and lifecycle planning was not a critical part of the process or its application architecture.

<sup>16</sup> [Microsoft IT cloud computing strategies continue to evolve](#). Microsoft. Accessed 31 March 2020.





Perhaps a one-off, bandaid application that didn't need to be used after the issue was fixed.

- Applications that simply duplicate functionality that is built into off-the-shelf or newer solutions more efficiently.
- Black box applications—you know if you turn them off bad things happen, but you don't know why or how.

Roughly 15% of applications in a portfolio assessment can generally be replaced by an off-the-shelf SaaS or third-party solution (according to Microsoft as well). For example, moving from Office servers to using Office 365 or internal portals to Sharepoint. This is pretty straightforward. No longer someone needing to watch an exchange server making sure it's overloaded. Not that you have the right powerpoint installed.

Customized and complex line-of-business applications generally consist of 50% of the portfolio.<sup>17</sup> Within that, there will generally be multiple sub-tiers of immediacy with most being "first to move." There will also be a percentage of applications identified as hard or costly to move—such as a legacy source code depot and associated toolchains.

A small percentage of applications will likely be determined to be more beneficial to remain on premises. This is usually less than five percent of the total portfolio. These are mostly applications with a defined sunset date where it does not make sense to invest any more time or energy.

## Financial Analysis

To build a business case for moving systems to the cloud you must also solve one important question: "how much will it cost us?" As an organization, you will want to perform a thorough analysis to determine this answer.

1. First, determine your current IT costs. While we know this isn't an easy task, it's important to understand the whole financial picture of your application's as-is costs should nothing change. This includes items like hardware, software, employees used to service equipment, overhead costs related to the facility, future investments necessary to continue the status quo, etc.
2. Second, find estimated yearly costs to run your applications in the cloud-based on necessary storage, users, processing power.
3. Lastly, calculate the costs associated with the migration process. This includes administration related to managing the process, employee time needed to test applications, and hiring consultants with an expertise in cloud migration to ensure best practices are used.

---

<sup>17</sup> [Microsoft IT cloud computing strategies continue to evolve](#). Microsoft IT Showcase. Accessed 2 April 2020.



Once you have these questions answered, you will have a better handle on your IT operating budget. It will be easier to predict and manage, hopefully keeping the accountants off your back and helping you build a case for innovation.

## Prioritize

Using the information discovered in the assessment, your organization can identify the candidates for modernization, the roadmap and its alignment to the business, and the budget required for modernization.

It's tempting to begin moving individual applications to the cloud immediately. Not to mention, cloud vendors make it easy to get started with semi-automated migration and modernization tools. However, with the right approach, it is possible to modernize a portfolio of applications in a way that yields value quicker and at a lower cost. In turn, this makes it easier and less expensive to stay current as products and technologies continue to evolve.

Remember, the current set of assets were most likely built or acquired over a long period of time. In most cases, applications were implemented to meet a tactical need where lifecycle planning was not a critical part of the process or its application architecture. We do not want to repeat any mistakes of the past.

Using the assessment of your portfolio as a whole you can also build a template for future use that aligns to individual apps, locations, or groups within your organization.

<b>First To Move</b>	<b>Next To Move</b>	<b>Hard or Costly To Move</b>
<ul style="list-style-type: none"><li>• Basic Web Apps</li><li>• Standard Three-Tier Applications</li><li>• Advanced Portals</li><li>• Any New Solutions</li><li>• Any Re-Architected Solutions</li></ul>	<ul style="list-style-type: none"><li>• High I/O OLTP</li><li>• High Business Impact Applications</li><li>• OLAP Systems</li><li>• Regulatory Apps</li></ul>	<ul style="list-style-type: none"><li>• Legacy Source Control Depot</li><li>• Associated Toolchains</li><li>• PKI Systems</li><li>• HVA Systems</li></ul>

Figure 4: Generalized Modernization Priority

Start with applications that are expected to have few dependencies to get your migration moving quickly.

Where we have seen a variance is which types of applications are the first ones to move to the cloud. The actual percentages seem to vary from organization to organization depending on their business needs. In general, the first applications enterprises move to the cloud are basic web applications or standard three-tier applications. These are certainly good candidates to start with.



## Micro Migration Plan: Application Level

Taking an existing application and migrating to any cloud offering is a serious undertaking. Enterprise applications can power entire business units, so any change to them should be well documented, communicated, and understood. As stewards of applications whose functionality helps business units make revenue, it is important that every part of the application's leadership, from the top down, understand what is being proposed when migrating an application to the cloud.

There are many steps that need to happen before the migration can begin. Every organization is different, so details will generally need to be customized.

### Step One: Define Workgroup

This step in the migration should be to define a workgroup of concerned development stakeholders of the application. These should be architects, team leads, senior developers, QA leads, DevOps, PROD support, scrum masters, product owners, and management that are currently involved in the day-to-day development and operations of the application.

The purpose of the workgroup is to define and drive the Migration Plan of the application to the cloud, including project scope, stakeholders, budget, constraints, dependencies, and risks.

### Step Two: Application Analysis

Individual applications must be understood in their current states before the best cloud approach can even be determined. The following are some of the top areas to review and be aware of at the individual application level before ever moving to the cloud.

#### State of Application as a Whole

When does the application get the most use and how much does usage fluctuate over time?

What is the total cost of the application to maintain in the current state and how does that compare to cloud?

Is there a compelling event or time frame driving the change?

What architecture documentation exists? Is it up-to-date?

Are there off-the-shelf software alternatives that could accomplish the same thing for the same or a lower cost?



Are the applications ready to move to the cloud?

What security mechanisms are in place? Examples: identity and access management, compliance, firewalls, endpoint security, etc.

What possible vulnerabilities will need to be accounted for in the new cloud architecture?

## State of Data

What is the inventory of servers, databases, and storage footprint?

Do you have a defined domain bounded context?

Will you host your data in the cloud? If not, are you prepared for hybrid implementation?

What architecture documentation exists? Is it up-to-date?

Do your apps access their data directly, or go through a service/abstraction layer?

Are you event-based, OLAP/BI, or highly transactional OLTP?

Do you have clean data to start with?

What access management is required?

Are there any security, privacy, or regulatory requirements that affect how data is stored and interacted with?

Are there any privacy concerns to move the data to the cloud?

## State of Application Code

What technologies and versions are used?

How are changes to it developed and deployed?

Are applications/services stateful or stateless?

Is your code brittle or highly dependent on other applications?

What metrics are available on the current system to measure performance and capacity?



What external dependencies do the services have? Hint: craft “dependency maps” of your applications.

## State of DevOps

Do you have a CI/CD pipeline in place?

Can you deploy parts of your codebase independently, or does everything have to go at once (including data changes)?

What automation is already in place—build automation tools, configuration management tools for installing application packages, orchestration tools for application lifecycle management, or automated testing?

Are you using a source code repository?

Are you using any automated scripts to deploy applications in prod and non-prod environments?

## State of Infrastructure

What current infrastructure is used? (Storage, compute, network, data collection, on-prem hardware—power, space, cooling—and operating environment, etc)

Analyze data flow exchanges between servers.

What is the total cost of the current IT infrastructure?

Will you be able to support cloud applications internally, or will you explore managed services?

Which cloud provider is a better model for your type of workload?

Will you use IaaS, PaaS, or SaaS or all (Hybrid)?

What are your primary business goals and objectives for your infrastructure migration to the cloud?

## State of Team Maturity

Do you have the right team members with the needed cloud technology experience for the migration?

Do team members have the necessary knowledge to set standards and expectations for cloud governance?



Do team members have a preference for a cloud provider?

Do you need to bring on additional staff or consultants to help adopt cloud computing technology?

Are the business stakeholders on board with a move to the cloud?

Do we have the skill-set to implement and maintain the security in the cloud?

These questions and their subsequent answers will help your organization determine your cloud readiness and establish the groundwork needed for migrating to the cloud. The goal is to make an educated move by understanding the application's specific requirements and state to choose the best path.

## Step Three: Develop Migration Plan

Using the information gained in step two, your Workgroup will then need to work together to create the Migration Plan. It includes a series of steps that need to occur for the application to be migrated to the cloud. The Development Workgroup's primary purpose is to define and drive this migration plan.

A key part of this migration plan is knowing each step of the migration process, and that includes analyzing and understanding what provider and models are best for your needs. This is explained in more detail in the future section [Assessing Cloud Providers](#).

The exact Migration Plan will differ from organization to organization, application to application. There are, however, some steps of the plan which every large organization will need to include or at least consider in the Development Workgroup meetings. It will first need to be defined and then will need approval from the stakeholders.

Defining the Migration Plan will have many shapes and forms, and will undoubtedly evolve over time. The following are the high-level aspects of the Migration Plan that should be considered.

- Application Model
- IP (Intellectual Property) Model
- Deployment Model
- Development Model
- Utilization Model—Azure or AWS, for example
- Governance Model
- Support Model



- QA Model
- Disaster Recovery (DR) Model
- Timeline
- Post-Implementation Success Verification

## Migration Plan Tips For Success

There are a number of areas we have seen “in the wild” that cause problems when not specifically defined in the Migration Plan. We have added additional details on these points as follows.

### Establish Management Tooling Strategies

Especially in enterprise-level organizations, a coherent cloud tooling strategy is required. More than just selecting the top vendors in a given category, it’s important to take the unique business requirements and apply it to cloud providers and offerings. The aim is to minimize the number of tools needed while fulfilling all management needs.

“The best strategy is a combination of solutions, based on the required degrees of cross-platform consistency and platform-specific functionality,” says Elias Khnaser, VP Analyst at Gartner.

“In all cases, organizations should prioritize the use of the cloud platform’s native toolset, augmenting that where needed with third-party cloud management platforms, cloud management point tools, DIY solutions and outsourcing,” Khnaser said.<sup>18</sup>

### Establish Governance Processes

Cloud providers offer on-demand resources with endless capacity. While extremely helpful with scalability, this makes it difficult for organizations to gain visibility into, and manage, what is being consumed.

Governance of cloud computing is challenging even with a single cloud provider involved and becomes additionally challenging as organizations move toward more complex implementations or a multi-cloud strategy. For every application in your organization, a strategy for governance and management of processes is key.

### Negotiate An Exit Strategy

To protect your company, you must ensure that you additionally prepare for an exit strategy. It’s kind of like a cloud prenuptial agreement. While it’s not ideal to plan for

<sup>18</sup> [6 Steps for Planning a Cloud Strategy](#). Gartner. Accessed 2 April 2020.



an exit before you even engage with a cloud provider, it's an important step to protect your company in case things change going forward.

If you read the fine print, you will usually find clauses that address renewal and termination. Often, termination can entail giving a 30-day written notice and contracts will automatically auto-renew.<sup>19</sup> It's important your organization understands the terms.

While you plan your implementation strategy, include an exit plan and potential costs of doing so. Should you need to switch vendors, it's important to know the process and costs associated with it.

## Step Four: Gain Acceptance From Stakeholders

With the first draft of the Migration Plan in place, the next step is to gain acceptance from the stakeholders.

Every application has two categories of stakeholders: horizontal and vertical. They are the parts of the organization that will be the most directly affected by every aspect of this migration, thus representing the first layer of approval. A list of groups should be defined within each category along with the purpose of the group and its leadership team.

With all of the stakeholders identified, the Migration Plan should specify a plan for engaging the stakeholders. The goal of the engagement plan is to gain approval from the stakeholders for the Migration Plan to proceed. Each stakeholder will have a different perspective on the migration effort based on their primary area of concern for the application.

### Common Stakeholder Questions

These are the pieces of the engagement plan that will be common among all stakeholders, so the Development Workgroup should have answers to the following questions immediately ready to provide to them:

- Why should the application be migrated to the cloud?
- Why should the application be migrated to the cloud provider chosen i.e. Azure vs. AWS Cloud?
- Why is the way the application is currently deployed or developed need to change?
- What are the benefits and risks?
- What is the timeline?

---

<sup>19</sup> [5 ways to avoid vendor lock-in](#). TechRepublic. Published 8 October 2018. Accessed 9 April 2020.





- What is the impact on my group?
- What will the day to day operations look like during and after the migration?
- What do you need from my group to make this migration a success?

At a common level, a meeting should be set up with the stakeholder’s leadership to obtain buy-in as well as to solicit advice and guidance. Each group’s acceptance of the migration is critical to the success of the effort as a whole, so every effort should be made to ensure the leadership of the group is confident their concerns have been met by the Migration Plan.

## Step Five: Implementation of Plan

Once your stakeholders have all approved the structured migration plan, you must then follow the steps you have laid out.

# Cloud Modernization Approaches

In general, there are six common approaches to consider when migrating or modernizing:

1. Rehost
2. Replatform
3. Rearchitect
4. Repurchase
5. Retain
6. Retire

Each of the six are known individually by many different names, but the information remains the same. For example, Rearchitect is also known as Refactor or Revise. But the concepts remain. Each is either a migration or a modernization approach.

## Rehost

“Rehosting,” otherwise known as “Lift and Shift,” is a relatively fast way to migrate applications from on-premises to the cloud. Essentially it involves lifting an application or landscape out of its current hosting environment and shifting it to another environment. Generally, this is re-deploying an exact copy of your application, database, and OS layer as-is from one environment to another—usually from on-premises to public cloud.

Because this approach requires no real changes to the application architecture or code, this strategy enables a faster and less disruptive migration process. Some



enterprise rehosting can even be automated (using tools like AWS VM Import/Export), although others prefer manually rehosting as a process to learn how to apply legacy systems to the new cloud platform.

Rehosting is the least expensive way for an organization to shift IT dollars from capital expense (CapEx) to operational expense (OpEx). The key with Migration is to consider the benefits and challenges with moving from a CapEx model to an OpEx model and to ensure that the application being migrated is not better suited for modernization. GE Oil & Gas, for instance, found that, even without implementing any cloud optimizations, it could save roughly 30 percent of its costs by rehosting on AWS.<sup>20</sup>

While Lift and Shift to IaaS is a viable and quick path to the cloud for many applications, this route will not unlock all of the possible cloud benefits. If it is done haphazardly it can result in an uptick in expenses. And, if an application runs slowly or inefficiently on-premises, it is unlikely to run any better on the cloud without modification. Legacy systems may have difficulties running outside of their current infrastructure and cause an increased demand for IT support.

Lifting and shifting legacy applications to a cloud environment may result in issues such as slow and unresponsive processing power. The application may need to use more processing power and memory in the cloud, negating potential cost savings. The Lift and Shift approach is typically easy to implement, but businesses must assess potential risks to ensure value and cost-effectiveness prior to enacting this approach.

## When To Rehost

- You want lower costs
- You seek to reduce CapEx
- You want to reduce risk
- You are on a deadline

## Rehost Advantages

- **Less Time & Labor Intensive:** this process is expedited when compared to other methods as you can leverage cloud features without worrying about the risk, complexity, cost, and time of a full refactor.
- **Lower Risk:** it does not include core architecture changes so there lower risk when compared with other approaches
- **Cost-Efficient:** cost savings compared to more expensive methods such as refactoring

---

<sup>20</sup>[GE Oil & Gas Case Study](#). AWS. Accessed 1 April 2020/



- **Use Of Certain Cloud-native Functionalities:** Apps can leverage the base cloud cost application. For example, replace common application components with a better cloud service, such as replacing Nginx in a VM with AWS Elastic Load Balancer.
- **Tactical Benefits:** like reducing the amount of time spent managing database instances.

## Rehost Disadvantages

- **Doesn't Fully Maximize Advantages Of Cloud:** Because rehosting essentially re-deploys an exact copy of your application (old architecture and all), it can be difficult to fully utilize cloud features like auto scaling and dynamic load balancing.<sup>21</sup>
- **Cost of Hosting:** While more cost-effective than on-premises, it can be more costly to run applications in the cloud this way than if you were to replatform or refactor.
- **Legacy Challenges Remain** - Applications with known issues before migration might suffer even more problems afterward, and legacy applications not optimized for cloud use are at increased risk to begin experiencing latency and performance issues.<sup>21</sup>

## Replatform

The Replatform approach is sometimes called “lift-tinker-and-shift<sup>22</sup>” As a cloud modernization strategy, this means making a few optimizations (cloud or otherwise) in order to achieve some tangible benefits. You aren't otherwise changing the core architecture of the application.

This is often considered a middle point—a level of action between rehosting and refactoring. Think the use of more managed services or alternative software licensing.

The design of your application will determine the number of changes you'll require during re-platforming. You may find that you need to break up a monolithic application into multiple applications and swap out components to get the most out of the move.

This category also includes modernizing database systems or integrating with APIs to enhance application connectivity and functionality over the cloud by keeping the application's original purpose intact.

---

<sup>21</sup> [Migration Approach: Rehost, Refactor or Replatform?](#) NetApp Blog. Accessed 2 April 2020.

<sup>22</sup> [6 Strategies for Migrating Applications to the Cloud.](#) AWS Cloud Enterprise Strategy Blog. Accessed 1 April 2020.



Another example of this would be to migrate web servers running on-premises to AWS, and, in the process, moving from WebLogic (a Java application container that requires license) to open source equivalent Apache Tomcat. AWS states the example media company saved millions in licensing costs on top of the savings and agility it gained by migrating to AWS.<sup>23</sup>

## When To Replatform

- **Modification of applications is required anyway:** whether versioning, licensing, or fine-tuning
- **You seek more cloud benefits:** like scalability, elasticity, and cost-efficiency rather than just moving the application to the cloud.
- **Most apps are three-tier web applications:** Once you have reshaped one, you can leverage this standard to improve efficiencies in migration as you move forward.
- **You want to avoid post-migration work:** A slew of tasks must be completed after migration to realize the full potential of the cloud. It sometimes makes more sense to make the changes in the application during the migration itself.

## Replatform Advantages

- **Less Effort and Commitment** (when compared to Re-Architecture): Replatforming allows you to experiment and make moves on a smaller scale without having to commit to an extensive modernization project.<sup>24</sup>
- **Ability To Leverage Some Cloud Benefits:** With a replatform, *some* cloud-specific functionalities like application scalability, infrastructure as code (IaC), and more are available.
- **Cost Efficient:** Typically, costly, large-scale development projects are not required when replatforming.<sup>23</sup>

## Replatform Disadvantages

- **Time Intensive:** when compared to Lift and Shift.
- **Risk of “Scope Creep”:** Without careful management and a watchful eye for unnecessary change, a simple refactor can easily become a large-scale, extensive Re-Architect.<sup>23</sup>
- **Automation is Required:** Replatforming can do very little and the benefits are limited without some form of automation. If you prefer to manually manage your workload in the cloud, replatforming may not be for you.<sup>23</sup>

<sup>23</sup> [6 Strategies for Migrating Applications to the Cloud](#), AWS Cloud Enterprise Strategy Blog. Accessed 1 April 2020.

<sup>24</sup> [Migration Approach: Rehost, Refactor or Replatform?](#) NetApp Blog. Accessed 2 April 2020.



## Rearchitect

There are a plethora of names for this approach: Replatforming, Rearchitecting, Redesigning, or Rebuilding. Regardless of which name you choose, it is essentially changing how the application is architected and developed, typically for it to have cloud native features.

If you seek the most from the cloud and tap into advanced capabilities like improved resiliency, global scale or maximum agility, cloud-native applications are built from the ground up and optimized for cloud scale and performance.

This approach tends to be one of the most expansive and expensive approaches to move to the cloud, yet it also provides the most opportunities to actualize cloud capabilities such as increased business agility, global scale, and improved resiliency of applications and services.

This approach is typically driven by strong business needs to add features, scale, or performance that would otherwise be difficult to achieve in the application's existing environment or current constraints. Triggers include wanting to deliver applications and features faster to users and customers, addressing security issues caused by legacy or outdated code, or enabling new business opportunities with innovative applications.

Design all applications so that they can easily be decoupled from the underlying infrastructure or platform of a hosting vendor. This reduces vendor lock-in and simplifies the task of transporting applications and data to an alternate vendor.

### When To Rearchitect

- **You want the highest level of cloud benefits:** Refactoring is the best choice when there is a strong business requirement of appending features, scale or enhanced performance.
- **You want lower long-term costs:** While initial investment is the more expensive approach, Applications can be optimized to operate at lower costs once in the cloud.
- **Your application has too many dependencies:** In that case, you may have no other option but to decouple and refactor your monolithic application.

### Advantages of Rearchitecting

- **Opportunity:** to redesign and leverage modern tools and technologies.
- **Long-term cost reduction:** there will be a reduction of costs over time resulting in higher ROI as compared to the less cloud-native applications.



- **Increased Resilience:** the application inherits the resilience of the cloud through decoupling
- **Increased Scalability:** applications can leverage the auto-scaling features of cloud services that scale up and down as per demand.

## Disadvantages of Rearchitecting

- **Vendor Lock-in:** The more cloud-native your application is, the more tightly coupled it is to the specific vendor's cloud you are in.
- **Skills:** Refactoring demands the highest level of application, automation, and cloud skills and experience to carry out the process.
- **Increased Time & Effort:** As refactoring is the complicated method of migrating from a non-cloud application to a cloud-native application, it can consume much more time.
- **Potential For Increased Mistakes:** You are essentially changing everything about the application, so there can be a higher risk of mistakes made.
- **Higher Initial Cost:** most or the part of the application must change, so upfront costs are higher

## Repurchase

Repurchasing is sometimes referred to as “Drop and Shop.” This method involves throwing out existing applications in favor of outsourcing one or more applications to a cloud-based software company (SaaS).

This is a great opportunity to reevaluate any contractual agreements to decide whether to retire an older application in favor of buying a newer updated version. You can easily research only solutions already built for the cloud platform you plan to migrate to, as opposed to migrating an older legacy application to the cloud where it may not be supported.

An example might be to replace your local database with a managed option such as Cloud Datastore, Cosmos DB, or Dynamo, or replacing an internal messaging tool with a tool like Slack.

## When To Repurchase

- When an application no longer satisfies business needs and a cloud-based counterpart does, which may also offer an improved feature set.
- When an on-premise tool also offers a cloud implementation, simply switch the manner in which your organization licenses the software to a service-based model.



## Repurchase Advantages

- Avoid IT development costs
- Low overhead with no infrastructure to manage
- Quick to implement
- Minimizes amount of services and applications that you need to manage
- Eliminates the need for you or your team to spend time on maintenance or updates

## Repurchase Disadvantages

- Minimal data access and customization
- 100% of all old functionality may not be available

## Retain

As mentioned during Application Portfolio Assessment, you will likely find a small percentage of applications that will be determined to be more beneficial to remain on premises, as-is, without changes. This is usually less than five percent of the total portfolio.

## When To Retain

- Applications with a defined sunset date where it does not make sense to invest any more time or energy,
- There is a plan to migrate to a vendor-supplied SaaS solution at a later date.
- An on-premise application the business is heavily invested in and/or has currently active development projects.
- Something still in the process of depreciation that you must retain to financially even
- Applications that must remain as-is to comply with regulations or security requirements
- Legacy OS and applications are not supported in the cloud

## Retain Advantages

- Zero effort
- The application stays as-is

## Retain Disadvantages

- The application stays as-is with no cloud features

## Retire

To justify migration, a product, service or application needs to hold a certain amount of value to your organization. Modernization provides a great opportunity to 'clean



house' by retiring or re-sizing applications that are no longer used or no longer valuable.

Often, applications in this category should have already been phased out long ago. The fact they are still running is a “surprise” to the organization. Turning them off generates immediate cost savings - both in the use of the application and in its governance.

As mentioned in previous sections, we have found that usually about 30% of a large enterprise's application portfolio could simply be retired.

## When To Retire

- It's a siloed departmental applications that runs under a desk somewhere
- Applications outside of central control
- Application duplicates functionality that is built into off-the-shelf or newer solutions also in use
- Application is no longer needed

### Advantages of Retirement

- Immediate cost savings
- Minification of application sprawl

### Disadvantages of Retirement

- Very few disadvantages

## Potential Improvement Points

No matter the current state of your application, there is always room for improvement in migration and modernization. There are a number of strategies that can help you better benefit from cloud computing.

### Using Serverless (FaaS) For New Features

When adding new capabilities to existing applications, serverless offers an attractive way to extend the functionality of an application without touching the actual source code which might be undocumented or outdated. Through this approach, existing code and functionality is left as-is and moved to managed cloud services. New capabilities are added incrementally using serverless functions triggered by events.

Instead of implementing new functionality in the main application, this approach suggests that you instead surround the existing application with pockets of new functionality using technologies like serverless functions that can be triggered via APIs and events. This approach is much easier and faster.





This also enables you to use advanced cloud-only capabilities like machine learning or AI with existing legacy applications that were built before such concepts were widely available.

Serverless computing is a cloud-computing execution model in which the cloud provider dynamically manages the allocation of machine resources. Pricing is based on the actual amount of resources consumed by an application.

This approach allows developers to focus on code without distractions – if they do not have to think about infrastructure, they will be more productive.

Serverless is optimized for any scenario where logic needs to be run in response to an event – and this definition encompasses a lot of application scenarios.

## Testing and Monitoring

Even if you only make small modifications in your application journey to the cloud, you still must conduct extensive testing. When you deploy on the cloud, you must monitor for an extended period of time. Pay particular attention to performance and user metrics to ensure correct configuration.

## Implement Container Technology

A popular option for quick, no-code application modernization is container technology. Containers are a packaging mechanism in which applications are abstracted from the environment in which they run. A common example is [Docker](#). This allows container-based applications to be deployed easily and consistently, regardless of whether the target environment is a private data center, the public cloud, or even a developer's personal laptop.

Containers are platform independent—build it once and run it anywhere. Containers run on any cloud, and even on premises, without changes. Containers also support all frameworks and technology stacks.

### **Benefits:**

- Containers support all frameworks and technology stacks across clouds and on-premises.
- No operating system overhead means you will have improved packing density and more resource efficiency.
- Faster application deployment with integrated CD/CD tools and orchestration.



## Rearchitect with Microservices

Microservices architecture means breaking large software projects into smaller, independent, and loosely coupled modules. A Microservice or Microservices architecture generally promotes more agile, scalable and resilient applications.

Individual services can be upgraded, changed or taken down without impacting the application. Developers work on individual services, which are smaller, easier to manage and understand. If individual services should fail, the application will continue to work without interruption.

When using a Microservices approach, decoupling the application concerns correctly into the right amount of serviced becomes the highest priority. You don't want a million services, but you don't want just one. Without this step, you simply end up with a bunch of microliths. Once decoupled, microservices can be developed, versioned, deployed, and scaled individually and separately from everything else.

When considering a monolith, deployment and scalability are usually two pain points. Deployment might be an 8-hour activity starting at 10:00 pm with a plethora of libraries validated manually. With microservices you split up into smaller pieces and automate deployment. Testing individual portions becomes a lot more feasible—not just faith-based testing and hoping.

This boundary also allows failure gates to be put in place to handle failure gracefully and allow these microservices to spin up and spin down without taking down the entire application.

Needy services can be scaled individually. For example, 50 versions of the same application handling calls in a round-robin fashion without having to scale up the entire app. One caveat is that it is easy to functionally decompose your whole application into services and end up with giant frankenstein. A happy balance of logical services is key.

## Design Application as Loosely Coupled

Your applications should be built or migrated to be as flexible and loosely coupled as possible. Cloud application components should be loosely linked with the application components that interact with them. Business logic should always be separated from the application logic, and that should be clearly documented. This will avoid the need to decipher business rules in case a migration to a new cloud provider would need to occur down the line.



We also suggest you abstract your applications from the underlying proprietary cloud infrastructure. This can be achieved incorporating REST APIs with popular industry standards like HTTP, JSON, and OAuth.

Not only does doing this reduce the level of lock-in to a single vendor, but it also gives your application interoperability that's required for fast migration of workloads and multi-cloud environments.

## Automation Architecture

In order to get ROI on your modernization effort moving to the cloud, Automation, also referred to as DevOps, is essential.

All cloud platforms provide automation features. When you are using a platform as a server PaaS, typically the platform implements the automation. A good example of automation offered by the Heroku PaaS.

A converged DevOps cycle provides the ability to execute on ideas quickly and iterate on feedback rapidly while at the same time maintaining the highest levels of quality.



# What Is The Best Approach?

Unfortunately, there is no absolute correct answer to choosing a migration or modernization approach. The reality is, any large-scale migration initiative will probably adopt several of the above scenarios. Different use cases require different solutions.

Figure 5: Pros and Cons of the 6 R's

In a perfect world, businesses would rearchitect every application to achieve all benefits from cloud computing. That's just not realistic for cost-conscious businesses, particularly those that have a vast array of applications in the portfolio.

Approach	Pros	Cons
<b>Rehost</b>	Fast, less resource-intensive, more cost effective when compared with on-prem	Higher risk, does not take full advantages of the cloud, potential performance issues
<b>Replatform</b>	Solid middle ground between rehosting and refactoring both in cost and cloud features	Scope creep is a real possibility, not all cloud features an option
<b>Rearchitect</b>	Build specifically to benefit from cloud native features	Most costly and labor intensive
<b>Repurchase</b>	Quick to implement, no development costs, cloud native	There may be life in your custom-built apps, and you'll lose future ROI.
<b>Retain</b>	Leave application as-is so zero effort	There may be life in your custom-built apps, and you'll lose future ROI.
<b>Retire</b>	Less time wasted managing unnecessary application	Custom-built application now with zero ROI

Picking the best approach for an application is a matter of finding the best that suits your specific needs. Start by checking if the application can be moved to a cloud environment in its entirety while maintaining cost and keeping operational efficiency in check. If the answer is yes, you may first consider the rehosting method.

Don't underestimate the amount of outdated, unused, duplicated, or over-provisioned legacy applications that you might find in your application portfolio. Identifying candidates for retirement or consolidation will generate immediate savings.

In general, we have found that simple workloads—customer-facing web sites, internal web portals, or standard 3-tier applications—are good candidates for PaaS. Managed cloud services for these workloads are so mature and reliable nowadays that often it does not make sense to go the IaaS route anymore. When it comes to custom applications, we suggest to our clients to move in stages:

- Move simple workloads directly to PaaS: web sites, static portals, standard three-tier applications



- Migrate complex solutions to IaaS first and gradually modernize later to unlock immediate benefits
- Keep legacy/undocumented code as-is and surround it with serverless functions to add features

For more complex solutions and workloads, one fast way to unlock cloud benefits is to first lift-and-shift as-is to the cloud, using VMs or container technology, and then gradually modernize where necessary.

You don't necessarily need to consider lift and shift as the end of the application's migration story. Very often, lift and shift can be considered a first stage in the larger migration plan. Applications can be migrated with lift and shift as-is. Then, once in the cloud, re-architected to take advantage of the cloud computing platform. In some situations, it is easier to re-architect applications while already in the cloud, as opposed to refactoring an application on-premise and then migrating it. This is partly because the cloud makes it easy to spin up realistic dev/test environments.

Re-architect is the most difficult and time-consuming, but also gives you the biggest opportunity to optimize the application and take advantage of cloud-native features. Repurchase is often the fastest and least expensive option in the short term.

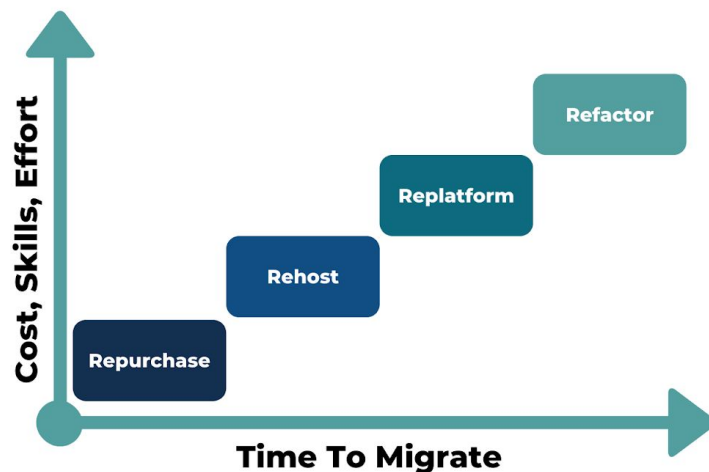


Figure 6: Comparison of Cloud Effort & Time

For undocumented or legacy code, our suggestion is to not touch it but instead surround it with pockets of new functionality using technologies like serverless functions that can be triggered via APIs.

A key facet of setting up your organization for success is to understand your existing cloud knowledge limitations. If you have the resources available in your organization who have been working with cloud-based solutions recently and can now shape applications for cloud compatibility, you're not out of the realm to consider using the Replatforming approach.

If your personnel do not have tangible cloud experience, you may need to consider alternatives. That could be:



- Rehosting while gaining education for successful future cloud implementations.
- Bringing in new employees or a consulting firm to help ensure re-architecting is performed using best practices and performing knowledge transfer.
- Choosing a cloud provider that, as a part of their service, gives your team the tools you feel will be sufficient for success.

## Assessing Cloud Providers

The field has [a plethora of competitors](#) in it, including the big three: Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP). We at Keyhole have used all three to bring huge success for our clients and they each have strengths and weaknesses. In this section we will briefly introduce the three leaders, then give areas to help you assess which might be the right fit for your organization.

- **Note:** We at Keyhole Software are consulting partners with both [Microsoft](#)<sup>25</sup> and [Amazon Web Services](#)<sup>26</sup>. However, as a technology-agnostic company, we seek to always use the right tool for a solution and intend to provide unbiased information.

At a high level, cloud platforms begin by providing a set of baseline services: compute, storage, networking, and database services. They all offer benefits like self-service and instant provisioning, autoscaling, plus features surrounding security, compliance and identity management.

On top of these basic services, each platform then builds other higher-level services that can be typically be categorized as one of five types:

- **Application:** Services designed to help optimize applications in the cloud.
- **Big Data:** Services designed to help process large amounts of data.
- **Machine Learning:** Services designed to help you incorporate perceptual AI such as image or speech recognition, or to train and deploy your own machine learning models.
- **Monitoring:** Services designed to help you track the performance of an application.
- **Security:** Services designed to keep your applications secure.

Do keep in mind the solutions mentioned are roughly comparable. Not every AWS service, Google Cloud service, or Azure service is listed, and not every matched service has exact feature parity.

---

<sup>25</sup> [Microsoft Competency Partner](#), Keyhole Software.

<sup>26</sup> [AWS Consulting Partner](#), Keyhole Software.



## Amazon Web Services (AWS)

- **Compute:**
  - IaaS: [Elastic Compute Cloud \(EC2\)](#); Selection of [EC2 Instance Types](#), [Dedicated Instances](#) and [Dedicated Hosts](#);
  - PaaS: [Elastic Beanstalk](#)
  - FaaS: [Lambda](#)
  - Containers: [Elastic Kubernetes Service \(EKS\)](#), [Elastic Container Service \(ECS\)](#), [Fargate](#)
- **Storage:** [Simple Storage Service \(S3\)](#), [Elastic Block Store \(EBS\)](#)
- **Networking:** [Virtual Private Cloud \(VPC\)](#)
- **Databases:** [Relational Database Service \(RDS\)](#), [DynamoDB](#)
- **Application:** [EC2 Auto Scaling](#), [Serverless Application Repository](#), [SNS](#), [CodeBuild](#), [Outposts](#), [Elastic Load Balancing](#)
- **Big Data Analytics:** [Kinesis](#)
- **Machine Learning:** [SageMaker](#)
- **Monitoring:** [CloudWatch](#), [X-Ray](#)
- **Security:** [Inspector](#), [Shield](#)

On top of the basic cloud computing services, there is multitenant block and file storage. These include object storage with an integrated CDN, Docker container services and event-driven “serverless computing”.

One of Amazon's biggest strengths is its dominance of the public cloud market—AWS has been the market share leader in cloud IaaS for over 10 years. It also has the most comprehensive network of worldwide data centers and a vast and growing array of available services.<sup>27</sup>

AWS strongly appeals to buyers seeking agile operations, but is also frequently chosen for traditional styles of IT operations. AWS is the provider most commonly chosen for strategic, organization-wide adoption according to Gartner.<sup>28</sup>

## Microsoft Azure

- **Compute:**
  - IaaS: [Virtual Machines](#), [Dedicated Hosts](#)
  - PaaS: [App Service](#), [Cloud Services](#)
  - FaaS: [Functions](#)
  - Containers: [Azure Kubernetes Service \(AKS\)](#), [Service Fabric](#), [Container Instances](#)

<sup>27</sup> [AWS vs. Azure vs. Google: Cloud Comparison](#). Datamation. Posted March 17, 2020. Accessed 6 April 2020.

<sup>28</sup> [Magic Quadrant for Cloud Infrastructure as a Service, Worldwide](#). Gartner. Published 16 July 2019. Accessed 6 April 2020.



- **Storage:** [Storage](#) ([Blob Storage](#), [Disk Storage](#), [Data Lake Storage](#))
- **Networking:** [VNets](#)
- **Databases:** [SQL Database](#), [SQL Server on Virtual Machines](#)
- **Application:** [DevOps](#), [Scale Sets](#), [Batch](#), [CycleCloud](#)
- **Big Data Analytics:** [HDInsight](#), [Data Lake Storage](#)
- **Machine Learning:** [Machine Learning](#)
- **Monitoring:** [Monitor](#)
- **Security:** [Security Center](#), [Advanced Threat Protection](#)

The second-largest provider is Microsoft Azure. In addition to standard computing cloud services, it provides many additional IaaS and PaaS capabilities including object storage (Azure Blob Storage), a CDN, a Docker-based container service (Azure Container Service), a batch computing service (Azure Batch) and event-driven “serverless computing” (Azure Functions).

According to Gartner, customers tend to adopt Microsoft who value the ability to use Azure to extend their existing investments in Microsoft technologies, or are interested in integrated specialized PaaS capabilities, such as the Azure Data Lake, Azure Machine Learning, or the Azure IoT solution accelerators.<sup>29</sup>

Microsoft tends to be a popular choice with organizations that have long-standing relationships with the vendor. So many enterprises already deploy Windows and other Microsoft software—Azure is tightly integrated with these other applications. As such, enterprises that use a lot of Microsoft software often find that it also makes sense for them to use Azure.<sup>30</sup>

## Google Cloud Platform (GCP)

- **Compute:**
  - IaaS: [Google Compute Engine](#), [Sole-Tenant Nodes](#)
  - PaaS: [Google App Engine](#)
  - FaaS: [Cloud Functions](#)
  - Containers: [Google Kubernetes Engine](#)
- **Storage:** [Google Cloud Storage](#)
- **Networking:** [Google Virtual Private Cloud](#)
- **Databases:** [Google Cloud SQL](#), [Google Firestore](#), and [Google Cloud Bigtable](#)
- **Application:** [Pub/Sub](#), [Object Lifecycle Management](#)
- **Big Data Analytics:** [BigQuery](#)
- **Machine Learning:** [AI Platform](#)
- **Monitoring:** [Cloud Monitoring](#)

<sup>29</sup> [Magic Quadrant for Cloud Infrastructure as a Service, Worldwide](#). Gartner. Published 16 July 2019. Accessed 6 April 2020.

<sup>30</sup> [AWS vs. Azure vs. Google: Cloud Comparison](#). Datamation. Posted March 17, 2020. Accessed 6 April 2020.





- **Security:** [Google Cloud Armor](#), [Encryption At Rest](#)

GCP combines an IaaS offering (Compute Engine), an PaaS offering (App Engine), and a range of complementary capabilities. This includes object storage, a Docker container service, and event-driven “serverless computing” (Google Cloud Functions).<sup>31</sup>

Google often stands out for its deep expertise around open source technologies, especially containers. This is in part to its integral role in the development of Kubernetes for orchestration and the Istio service mesh—which are quickly becoming industry-standard technologies.<sup>32</sup>

GCP initially appealed to adopters with demonstrated strengths associated with big data and other analytics applications, machine learning projects, cloud-native applications, or other applications optimized for cloud-native operations. It has also begun to attract enterprises with traditional workloads such as SAP.<sup>33</sup>

## Others

There are [hundreds of cloud providers](#) on the market. If we needed to make a short list, these would be the others we would add to the “top three” (in no order):

- [IBM Cloud](#): If we were comparing the top four cloud providers, IBM would be the fourth; it has one of the world’s most powerful AIs (Watson) in the cloud and offers IaaS, PaaS, and SaaS options. It lists American Airlines, Panasonic, and Honeywell, as customers.<sup>34</sup>
- [Oracle Cloud](#): Offers IaaS, PaaS, SaaS, and Data as a Service (DaaS) and is used by Zoom Video Communications<sup>35</sup>, Freddie Mac<sup>36</sup>, and Aer Lingus<sup>37</sup>
- [Alibaba Cloud](#): China’s primary cloud provider. It lists Ford, Air Asia, and KPMG as customers.<sup>38</sup>

## Assessing Providers

A critical step in ensuring a seamless cloud experience is ensuring the cloud provider or providers selected are a match with your organization’s needs.

<sup>31</sup> [Magic Quadrant for Cloud Infrastructure as a Service, Worldwide](#). Gartner. Published 16 July 2019. Accessed 6 April 2020.

<sup>32</sup> [AWS vs Azure vs Google Cloud: What's the best cloud platform for enterprise?](#) ComputerWorld. Accessed 6 April 2020.

<sup>33</sup> [Magic Quadrant for Cloud Infrastructure as a Service, Worldwide](#). Gartner. Published 16 July 2019. Accessed 6 April 2020.

<sup>34</sup> [IBM Cloud case studies](#). Accessed 7 May 2020.

<sup>35</sup> [Oracle wins cloud computing deal with Zoom as video calls surge](#). Reuters. 28 April 2020. Accessed 7 May 2020.

<sup>36</sup> ["Freddie Mac"](#). Oracle Marketing Cloud. Accessed 7 May 2020.

<sup>37</sup> ["Aer Lingus Revenue Per Email Flies 132% Higher Than Travel Industry Benchmark"](#). www.oracle.com. Accessed 7 May 2020.

<sup>38</sup> [Customer Success Stories](#). Alibaba. Accessed 7 May 2020.



## Service Levels

You know your business' needs, and your provider must provide a solution that meets your requirements for availability, response time, support, and capacity.

- Does your cloud provider have a reputation for reliability?
- What is their average uptime over the past three years?
- Do they use reliability safeguards like backup power sources and redundant servers?
- Where is the cloud computing provider located?
- Is the region's primary language English?
- What is the success rate of the provider meeting the contractual requirements of cloud migration projects?

## Support Models

Much of your support needs can be determined by your analysis of team maturity surrounding cloud services.

- What level of help do you anticipate your organization will need—does the cloud provider offer comprehensive support in line with your needs?
- If you need help, will you be able to get it quickly and simply?
- What type of support is offered - i.e. solely via chat, phone, email?
- Some cloud providers do not provide customer service to all users, but just administrators. What are your needs?

## Security & Compliance

It is important to determine that the cloud provider approaches security in a way your organization deems sufficient, or follows the regulations your organization is held to. There are three main areas of security focus:

- Data security, such as encrypting data, and controlling user access or privileges.
- Physical security, like the protection of physical assets at a geographic location.
- Infrastructure security, such as ensuring that security patches are updated as soon as possible and abnormal behavior is sought out.

It is also important to also have a clear contractual relationship, legally enforceable, between you as the cloud service customer and a cloud service provider. Ensure that if something goes wrong, you have a legal agreement that will help support your aims.

- What are the compliance requirements that the cloud service provider meets?
- What are the Disaster Recovery locations and the security measures available at these locations?



- What is the minimum level of data security that you would accept from a cloud service provider?

## Integration

When choosing a cloud provider, you should analyze how the cloud architecture will be incorporated into your workflows, both now and in the future.

For example, if your organization has already invested heavily in the wide world of Microsoft, it might make more sense to proceed with Azure to improve cohesiveness. Bonus: Microsoft gives its customer's licenses (and often some free credits). Likewise, if your organization relies more on Amazon or Google services, then it may be best to look to those vendors for ease of integration and consolidation.

- What technology companies are already providing services to your organization?
- What other development needs surround your move to the cloud? Perhaps you'd prefer a cloud provider that also integrates container technology automatically.

## Cost

While it shouldn't be the only priority, there is no doubt it will play a pivotal role in provider analysis. A number of different payment and subscription models exist. For example, the "big three" currently have the following models available:

- **Azure:** Azure bills customers on-demand by hour, gigabyte, or millions of executions, depending on the specific product. It also provides the option to reserve instances, like AWS, and price match other services.<sup>39</sup>
- **AWS:** Amazon determines pricing by rounding up the number of hours used with a minimum use of one hour. Instances can be purchased in one of three ways: Pay-as-you-go: Pay for what you use, no upfront cost; Reserve instances for one or three years, with an upfront cost based on utilization; and volume discounts, acquire more services and receive volume discounts for specific services, such as S3.<sup>40</sup>
- **Google Cloud Platform:** GCP bills for instances per second used. Interestingly, Google also offers "sustained-use discounts" and "committed use discounts" for computing services that offer a simpler and more elastic model compared to AWS's reserved instances.<sup>41</sup>

All offer calculators for estimating future cloud spending based on unique needs.

- [Microsoft Azure Calculator](#)
- [AWS Calculator](#)

<sup>39</sup> [Azure pricing](#). Microsoft. Accessed 2 April 2020.

<sup>40</sup> [AWS Pricing](#). AWS. Accessed 2 April 2020.

<sup>41</sup> [Do more for less](#). Google Cloud. Accessed 2 April 2020.



- [Google Calculator](#)
- [IBM Calculator](#)
- [Oracle Calculator](#)
- [Salesforce Pricing Information](#)
- [SAP Pricing Information](#)
- [Public Cloud Cost Comparison Calculator](#) - AWS, Azure, and Google Cloud

While the different approaches all have their unique differences, you must determine the best for your organization based upon your projected usage patterns. And remember that not all customers will pay the sticker price, especially at the enterprise level where volume discounts can be negotiated.

## Exit Strategies

In the case of changing business needs, what will occur should you need to end your relationship with the cloud provider?

- What terms for cancellation is your organization comfortable with?
- What are the conditions under which you can terminate the contract?
- Will you need the vendor to assist with a deconversion should you want to migrate to another vendor?

Often, these terms can be negotiated into a Service Level Agreement (SLA) up front on a case-by-case basis. But to negotiate, you must know what you're seeking.

## Conclusion

This document covered a myriad of cloud computing topics with an eye for information your organization must know for a successful cloud migration—from analysis and migration strategy to the selection of the right platform, deployment, and provider for your cloud needs.

This is a high-level overview of the topics discussed:

- History of cloud technology and what it means for business financials
- An introduction to the different strategies for computing services—PaaS, SaaS, IaaS, and FaaS.
- A look into deployment models your organization can consider including public, private or hybrid—each with associated pros and cons.
- Macro and Micro strategies for analyzing your current cloud readiness
- A general path for cloud migration with tips for success.
- A discussion of the 6 R's—the options to migrate your applications—with advantages, disadvantages, and sample use cases
- Tips for assessing a potential cloud providers with a general overview of the top three cloud computing platforms



When evaluating the move to the cloud, no cloud hosting option should be taken off the table too early—there is no one-size-fits-all solution. Some applications will be moved to IaaS, some will be redesigned to use PaaS services, and yet others are a good fit for replacement by a SaaS solution. What is the right fit for you might not be the right fit for other organizations.

We at Keyhole Software are here to help if you still have questions like the following:

- How can I integrate my existing systems with the cloud?
- Based on my industry and technology, is there a cloud service provider that is best for me?
- What is it going to cost to get to the cloud?
- Is my organization ready for the cloud? If not, what steps do I need to take to get there?
- How do I get more speed, agility, and performance for my existing IT assets?
- What type of “X-as-a-Service” is the best fit for my application?
- What backup and disaster recovery strategies should I have in place?
- How do I manage security?
- What types of cloud governance and standards do I need to set?
- Which application should be my first to take to the cloud?
- How do I ensure my staff has the right knowledge to migrate us to cloud computing?
- How do I control ‘shadow IT’, or ‘zombie apps’ or workloads that are consuming more resources than the value they are delivering?

Please reach out if there is anything we can do to assist your organization in your cloud migration journey.

## Further Research

- [Gartner Cloud Resources](#) by Gartner
- [AWS to Azure](#) services comparison by Azure
- [Google Cloud to Azure](#) services comparison by Google Cloud
- [Google Cloud to AWS](#) services comparison by Google Cloud
- [AWS Migration Whitepaper](#) by AWS Professional Services
- [Microsoft Azure Migration Journey Home](#) by Microsoft
- [5 Phases of Moving to Google Cloud](#) by Google Cloud
- [A Cloud Services Comparison Of The Top Three IaaS Providers](#) by VMWare CloudHealth
- [Amazon AWS: Complete business guide to the world's largest provider of cloud services](#) by ZDNet



## About Keyhole Software

Keyhole Software is a software development and consulting firm. Our expert employee consultants excel as “change agents,” helping our clients to be successful with technologies that bring competitive advantage.

Expert consulting is the core of Keyhole Software. Our cloud expertise spans strategic assessment and enterprise roadmapping, proof-of-concepts, migration, cloud-native development and implementation, and the education that your team needs to actualize the benefits from this technology.

We have proven our cloud expertise to earn partnership status with [Microsoft](#) and [AWS](#) and helped our clients in every industry to leverage the benefits of cloud technologies. Expert consultants at Keyhole Software can bring that same insight and knowledge to your cloud initiative. We consult nationally across the United States with clients in every vertical. The Keyhole Software corporate office is located in Lenexa, Kansas, just south of Kansas City. Additional teams are located in St. Louis, Lincoln, and Omaha.

Knowledge transfer is a priority of the Keyhole Software team. As such, we often host [educational events](#) and author [white papers](#), [videos](#), and [weekly technical blogs](#). Additionally, we have a number of open source implementations available publicly on our team [Github](#).

### Related Services Snapshot

- **Cloud Consulting:** including provider-agnostic, [AWS-focused](#), and [Azure-focused](#) services.
- **Cloud Readiness Assessment & Roadmapping:** Assessing current status and strategically planning the best path for successful cloud adoption.
- **Application Cloud Migration:** Migration of existing applications to PaaS or cloud offerings, including lift-and-shift, re-architecting, re-platforming, and others.
- **Cloud Solution Implementation & Development:** Quickly creating and deploying mission critical web apps and POCs that scale with your business.
- **Cloud Education:** Teaching your team to be successful with DevOps, orchestration, and cloud techniques required in a successful implementation.

### Contact Keyhole Software

**Company Website:** <https://keyholesoftware.com>

**Products Website:** <https://keyholelabs.com>

**Phone:** 877-521-7769

**Email:** [asktheteam@keyholesoftware.com](mailto:asktheteam@keyholesoftware.com)

**Headquarters:** 11205 W 79th Street, Lenexa, KS 66214