

Choosing the Best Cloud Provider(s) For Your Development Workloads

Summary:

A company's decision to move some or all its workloads to the cloud is often a difficult one to make, especially in overcoming any institutional fears of data loss or improper security setup. After making this decision, the next step is to evaluate the best cloud provider for the organization's development workloads.

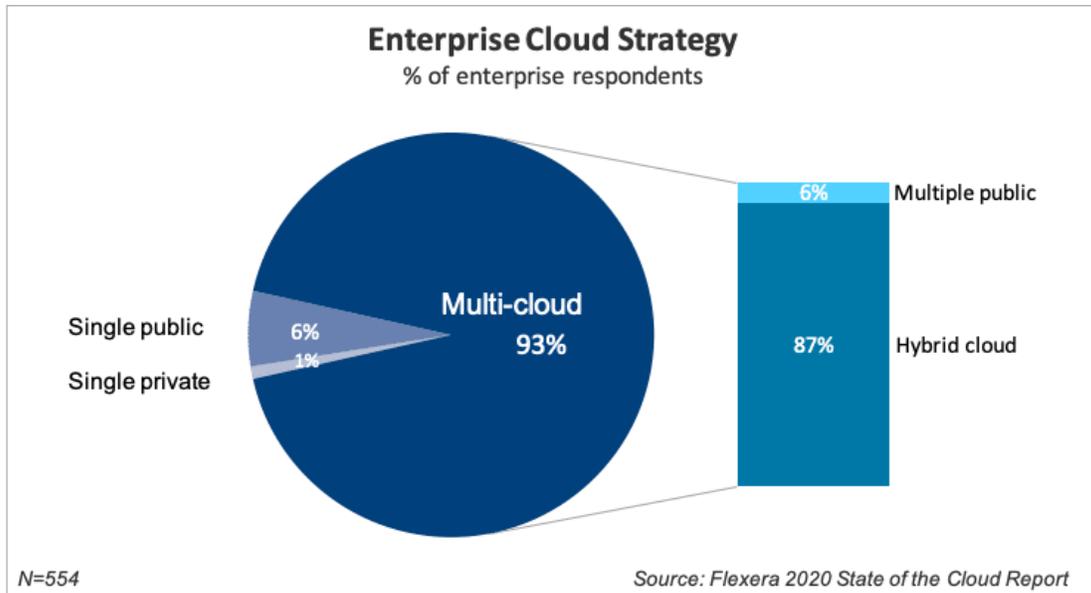
Once an organization has made the important decision to move solution development to the cloud, whether wholesale or simply starting a Proof of Concept (PoC), the next major step is to evaluate which cloud provider(s) have the capabilities and features your development organization requires. This analysis should be driven by the type of development features available, current or proposed coding languages, and any cloud services already being utilized (e.g., email, productivity applications, business intelligence, etc.). Once completed, this analysis can become an enterprise development guideline detailing out the best cloud platforms for different types of implementations.

For cloud development solutions to support an organization's Software-as-a-Service (SaaS) based application integrations, an enterprise-class provider is needed as the backbone of the many REST-based API data transfers needed. These workloads may support data integrations between core applications and subsidiary systems that reside in either the cloud or on-premises, as well as any future API-based applications that may be developed.

After documenting the organization's current and proposed development requirements, including whether to utilize cloud Infrastructure as a Service (IaaS) or Platform as a Service (PaaS) models, a review of any current cloud provider usage in the organization should be done. One of Hylaine's consultants recently published a great article around Cloud Migration Strategies on existing applications (<https://www.hylaine.com/news/cloud-migration-strategies>).

As an example of possible discoveries, the usage findings may show that one provider is already used for authentication and office productivity applications, where another may be currently utilized for server hosting or Big Data storage. Expanded usage of an existing cloud provider may allow for economies of scale, improved resource usage, and lower costs than other providers.

This type of usage with multiple cloud providers is commonly known as a Multi-Cloud strategy and is an approach for some enterprises. This allows them to be cautious of vendor lock-in and gives the ability to abstract applications so they can be migrated across cloud platforms if needed using common development languages or containerization. This mindset also helps with resilience as all cloud providers suffer outages from time to time, thus spreading solutions across several clouds can help lower the risk of all mission-critical applications becoming unavailable. (Flexera - 2020 State of the Cloud Report: <https://www.flexera.com/blog/industry-trends/trend-of-cloud-computing-2020/>)



One thing to note on a Multi-Cloud strategy is that by spreading out usage and/or applications between multiple cloud vendors, enterprises may miss out on potentially enormous cost savings by fully utilizing just one platform. Organizations should use cloud-agnostic tooling (e.g., containers, supported data engines, etc.) and then choose primary and secondary clouds, with the latter usually only for Disaster Recovery (DR).

While utilizing multiple cloud providers can provide many benefits such as cost and resilience, for developing specific solutions it is important to select the best platform that matches the integrations needs. Gartner's Magic Quadrant for Cloud Infrastructure and Platform Services is a well-known agnostic review of the core players in this space. It is a good start for reviewing which provider is leading and in which categories.

(ZDNet.com - Top cloud providers in 2020: <https://www.zdnet.com/article/the-top-cloud-providers-of-2020-aws-microsoft-azure-google-cloud-hybrid-saas/>)

Figure 1. Magic Quadrant for Cloud Infrastructure and Platform Services



After reviewing the categories top providers are leading in, it is important to then dive deeper on specifics for development workloads when it comes to creating a good analysis. There are several development usability and management factors that should be considered that are not evaluated in a magic quadrant analysis.

1. Integration with other platform services and enterprise infrastructure:
 - The provider(s) should be able to integrate well with on-premises network capabilities.
 - Identity and security management is critical and should not require constant monitoring or changes to keep working with other systems.
 - Other provider platform services should have the capability for native integrations when needed without having to go outside that platform.
2. Services and programming languages that match application needs:
 - Application features should be matched to components needed such as serverless functions, scheduling, queue management, and other API-related needs.
 - Programming languages for the implementation should be matched against most current technologies and team skillsets.

3. Ease of creation and management of deployed services:
 - Friction-less deploying of services and support components without having to know hidden exceptions or guesswork.
 - Ability to easily manage created services from the standpoint of resource usage, current state, and cost analysis.
 - 'Single-pane-of-glass' for viewing resources and features to monitor proactively for bottlenecks and issues.
4. Toolsets and management capabilities:
 - Provider(s) should have native Continuous Integration / Continuous Delivery (CI/CD) toolsets or first-class integrations with major third-party CI/CD vendors that can manage work tracking, source code version control, and automated build releases.
 - Coding toolsets should be able to build what is necessary easily and integrate with above mentioned CI/CD systems.

With these above guidelines, putting together a cloud platform recommendation that fully supports development efforts will be possible. These development needs cover how the team tracks and manages work, ensures that all codebases are stored with proper merge/branch methodology, that the code is always built and tested independently, that solutions are automatically deployed to the right service, and that all of this can be setup and managed in an efficient manner.

Taking the above guidelines, we can now create a succinct list of features an enterprise-class cloud platform should provide to the organization.

- Core services and features that match components needed in the layer (i.e., API's, Scheduling, etc.), as well as the programming language and development frameworks needed.
- Project management task tracking solution that is integrated within the platform.
- Source code repository for version control that links to task tracking solution above to match requirements to development time needed and taken.
- Native CI/CD system for automated builds, testing, and deployments to core services of above source code repository.
- Ease of deploying all the above needs and integrating them together to focus more on development.
- Ease of maintenance and monitoring of all aspects of the platform including costs.

Having defined how to analyze an organization's development workloads and what is needed in a cloud provider to support those efforts, a comparison of the top three cloud platforms can be accomplished. The following sections detail out the current top three providers pros and cons based on our above development factors. From there a grade can be given on how well they meet these important guidelines.

Amazon Web Services (AWS)

AWS is a public cloud computing platform offering compute power, database storage, content delivery and other functionality.

Pros:

- Over 175 services from data centers globally.
- Largest IaaS provider of all three cloud providers.
- Pay-as-you-go approach for pricing.

Cons:

- More individual services need to be deployed to create a working CI/CD chain due to their separations.
- Some services do not innately connect to each other, or specific hand-typed information must be entered from other services, thus slowing down deployments.
- Training for employees may be increased as there is extra platform knowledge that needs to be transferred along with component integrations.
- Annual costs for utilizing serverless capabilities shows to be the most expensive of all platforms.
- Enterprise level support must be purchased separately.
- EC2 has its limitations, but can request upgrades depending on region and usage requirements.

Grade:

- B

AWS Pricing Calculator by Service:

<https://calculator.aws/#/>

Azure

Microsoft Azure is a public cloud computing platform with solutions including IaaS, PaaS, and Software as a Service (SaaS) components that can be used for services such as development, analytics, virtual computing, storage, networking, and more.

Pros:

- Easily deploy, integrate, maintain, and monitor all services utilized within the platform including cost analysis.
- Use Azure DevOps which fluidly handles project management, source code, and CI/CD to services all in one system without having to couple together multiple disparate toolsets. It also integrates well with many other platforms and clouds.
- Works with other Microsoft platforms and services, such as Microsoft 365, and for future business / global initiative integrations.

- Training on services deployed to employees may require less time due to the abundance of online resources.
- Annual costs for utilizing serverless capabilities shows to be less than the other two platforms.
- Pay-as-you-go approach for pricing or purchase ahead.
- If a Microsoft customer already, Azure is an included component.
- Integrations include Windows Server and Linux Virtual Machine.
- Strong presence in the enterprise community.
- High availability and redundancy in data centers on a global scale.

Cons

Licensing and costs can be confusing if not already obtained at the enterprise level.

Grade:

- A (Would be an A+, but licensing requires a lot of work sometimes)

Azure Pricing Calculator by Service:

<https://azure.microsoft.com/en-us/pricing/calculator/>

Google Cloud (GCP)

GCP is a public cloud platform whose services are delivered to customers on an as-you-go basis, by way of service components. Its primary service model is based around the development and deployment of more modern, containerized applications.

Pros:

- App Engine is the core application development tool, with many common languages supported.
- Native Kubernetes container support.
- Emphasis on big data tool suites.
- Storage is fast with uncapped bandwidth and strongly consistent listings. BigQuery, BigTable and many other services have "no-ops" so there's no overhead.
- Workloads for OCR (Optical Character Recognition) are currently very price competitive as it works to increase market share.

Cons:

- There is no inherent project management tracking tool that integrates with the source code or CI/CD toolset to track work.
 - A hosted BitBucket instance would be needed, which has the capability to deploy to GCP.
 - A hosted Jira instance would be needed to connect with BitBucket for issue tracking and workload management.

- The platform does not support as many modern programming languages as Azure and AWS.
- Annual costs for utilizing serverless capabilities shows to be the 2nd most expensive of all platforms.
- Google's current development culture has led to a disproportionate number of retirements and platform obsolescence, often causing application rewrites when there is breaking changes in functionality.

Grade:

- C- (Admin privileges are required to do full development, and the use of Jira / BitBucket for change management)

Google Cloud Pricing Calculator by Service:

<https://cloud.google.com/products/calculator>

In defining an organization's development cloud strategy, the team must focus not only on its current solution development needs, but also with an eye towards future tools that an enterprise-level platform can provide. While the three evaluated platforms can handle most any development workload, there may be significant drawbacks that are specific to one organization over another due to existing development resources or additional integrations needed.

Hylaine is a cloud and technology agnostic partner and many of our engagements often consist of helping customers complete this type of analysis. Whether for new enterprise rollouts, reviewing existing cloud implementations for performance improvements, or looking for cost savings. Our experienced team can help partner with your organization to make this journey to the cloud as frictionless as possible.