*Review*

# Representative Service Providers and Their Selection in Cloud Computing Domain; A Comprehensive Overview

**Pranish Dahal[1]\* and Ritika Prasai[2]**

[1]   Department of Science and Technology, Pokhara University, Nepal
[2]   Texas Institute for Applied Environmental Research, Stephenville, Texas, USA
\*Corresponding author: pranish.dahal.np@gmail.com

**Abstract:** The popularity of cloud computing is growing owing to its large data storage capacity and high computation power. It provides online, on-demand, scalable application solution, removes hardware and software barriers for non-specialist, rapidly integrates and deploys desired and necessary facilities, supports quick upgrading and addition of features. Users get benefitted with the selection of the appropriate cloud computing platform for their projects. Here, our paper provides a comprehensive overview of the services provided to the users by the most common cloud computing service providers. This paper could be used as a reference while selecting the best service provider based on the requirements of the projects.

**Keywords:** cloud computing; data storage; users; service provider; software; hardware

## 1. Introduction

The underlying concept of cloud computing has a long history. It provides online, on-demand, scalable application solution, removes hardware and software barriers for non-specialist, rapidly integrates and deploys desired and necessary facilities, supports quick upgrading and addition of features. It provides services like data storage, computation power including software, hardware, or platform. Service provided through cloud computing can be public, private, or hybrid. The particular company providing cloud computing services to its customers/users/clients is indicated as service provider in technology domain. Therefore, we will also use the term service provider to refer those company and customer/clients as the user in this paper. We discuss about the service provided to the user by the service provider in this paper. There are multiple reasons for the positive inclination of people towards service providers. The prominent being their cost effective features. Service providers on demand facilitates to add or remove required resource for the application development. This makes the user flexible enough to use and pay only for the most necessary and used resource and does not force the user to buy the physical computing resources [5]. In case of need, users can utilize the hardware/software/server resource provided by the service providers at any point of time. In addition, the platform provided by the service providers has security functions enabled by default. They follow industry standard security protocols and also keep updating the service (like Database) with the latest patches. Users can also utilize the extra security facility services for the application [6]. Furthermore, users can have full control over the services. They can either scale up the resource or scale down as per the need of the application. Users need not worry about the physical changes required while up scaling or down scaling the services. One example of scaling is to increase or decrease the storage size of computing instance.

Software service (SaaS), Platform service (PaaS), and Infrastructure service (IaaS) are some of the most commonly provided services by the service provider. When the service provider is providing software services to its user, the services can be utilized to develop, improve, and upgrade the software application for user's own usage. Although there are challenges [2], users can focus on the business process rather than the software

development process using this service. Under platform service, the provider provides a platform to develop, maintain and test the application developed using the services. Users can focus on applications rather than maintaining servers or networks. Providers provide servers, virtual machines (VM), and operating systems for usage to the user when providing the infrastructure services. VM is like computing device inside a host server which has its own operating system [3]. The host server can be fully utilized when many VM is operated under the same host server. The user needs to pay as per the usage of the service.

Another service provided by service provider is serverless architecture [11]. Here, the application of the user is hosted by the service provider in the cloud. This architecture eliminates the process of managing hardware or software platform and is executed when necessary. Only the functional part of the application is required for execution. This increase the focus of application in business needs and also may decreases the application development time. Although most of the serverless architecture have execution limitation, using this architecture, development cost can be reduced since it only runs when required without any other costing server. To utilize the services provided by the service provider, user can use a user interface, command-line options, application programming interface or system development kit provided by the service provider. The service provider have proper documentation, best practices and multiple approaches to develop application in their platform.

### 1.1. Selection of the service provider

Selection of the service provider depends on many factors (Figure 1, Table 1-8). Most common factors being cost, scalability, development time. Cost is the prime factor that determines the selection of the service provider in most of the cases [7]. Selection also depends upon the application requirement and usage [8]. Users can figure out the estimated cost using the cost calculator service available. Scalability is another factor that impacts in the service provider selection process. Deployment services, feasibility of scaling up/down the application hosted in the provider will decide the selection. Similarly, users think about the accessibility, easiness in pre-processing the data while selecting the service provider, users select no coding option if they want the service provider to support no code feature for their data processing. Users want to perform all the necessary operations be executed using drag and drop options. In addition, user's app/tool may not need high computing servers. Users can check if user's tools/apps can be executed in browsers without being dependent on servers and utilize the exact service of the service provider.
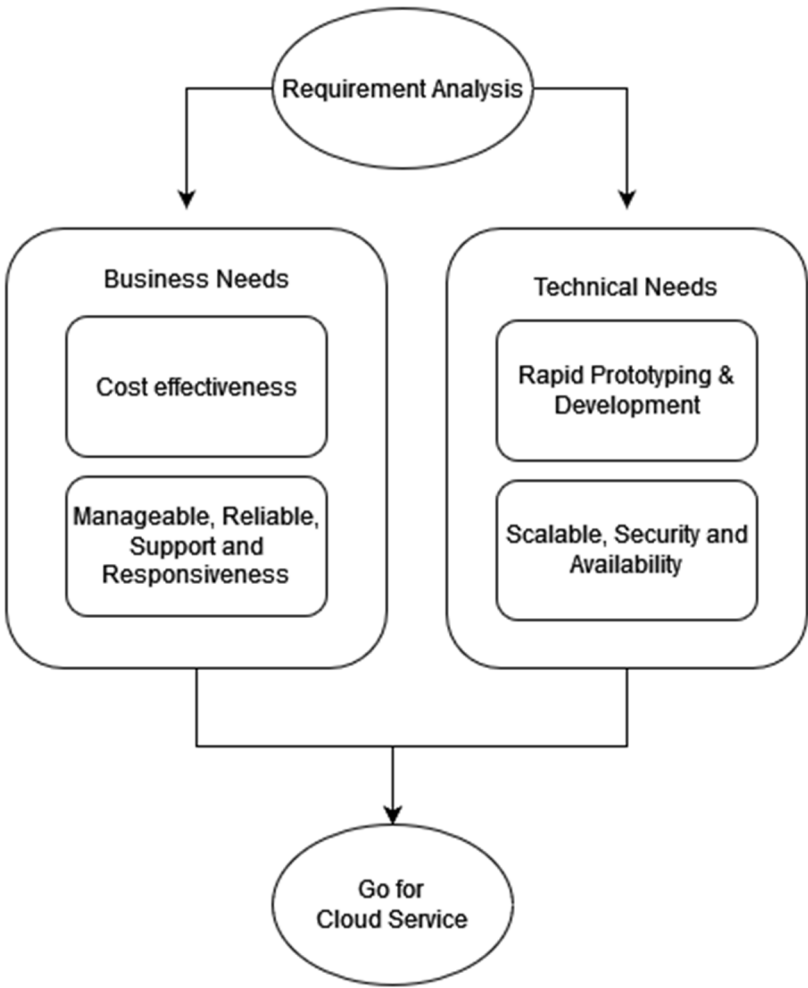
**Figure 1.** Different business and technical need to go for cloud service provider.

**Table 1.** Name of most popular cloud service providers.

| Name of Service Provider |
|---|
| Amazon Web Services (AWS) |
| Google Cloud Platform (GCP) |
| Azure |
| Digital Ocean |
| Linode |
| Alibaba Cloud |
| Oracle |
| IBM Cloud |

**Table 2.** Popular service provider based on the computing instance, file storage, serverless feature details. Service providers have not been ranked or kept in any order based on any particular feature.

| Service Providers | Computing Instance | File Storage | Serverless |
|---|---|---|---|
| AWS | Elastic Compute Cloud | Simple Storage Service | Lambda |
| GCP | Compute Engine | Cloud Storage, FileStore | Cloud Functions |
| Azure | Virtual Machine | Azure Files, Azure Blob Storage | Azure Functions |
| Digital Ocean | Droplets | Space Object Storage, Volumes Object Storage | Digital Ocean Functions |
| Linode | Virtual Machines (CPU Based) | Block Storage, Object Storage | Function as a Service |
| Alibaba Cloud | Elastic Compute Service | Elastic Block Storage | Function Compute |
| Oracle | Virtual Machine | Oracle Cloud Storage | Oracle Functions |
| IBM Cloud | Virtual Workloads | IBM Cloud Storage (File, Object, Block) | IBM Cloud Functions |

**Table 3.** Popular service provider based on their computing instance feature details. Service providers have not been ranked or kept in any order based on any particular feature.

| Computing Instance | Low Range Instance | High Range Instance | Self-Managed Service |
|---|---|---|---|
| AWS | 1 vCPU, 0.5 GB | 64 vCPU, 256 GB | Yes |
| GCP | 2 vCPU, 8 GB | 96 vCPU, 1360 GB | Yes |
| Azure | 0.5 GB, Balanced CPU-to-memory ratio | 128 vCPU, 3,800 GB | Yes |
| Digital Ocean | 1 vCPU, 0.5 GB | 32 vCPU, 256 GB | Yes |
| Linode | 1 vCPU, 1 GB | 32 vCPU, 192 GB | Yes |
| Alibaba Cloud | 2 vCPU, 8 GB | 104 vCPU, 768 GB | Yes |
| Oracle | 2 vCPU, 1 GB | 64 GB, 128 vCPU | Yes |
| IBM Cloud | 1 vCPU, 2 GB | 64 vCPU, 512 GB | Yes |

**Table 4.** Popular service provider based on their serverless feature details. Service providers have not been ranked or kept in any order based on any particular feature.

| Service Provider | Serverless Name | Running Time Limitation | Configuration | Supporting Language | Connectivity to Provider's other services? |
|---|---|---|---|---|---|
| AWS | Lambda | 15 Min. | Runs on Linux OS | Node JS, Java, Python, Ruby | Yes |
| GCP | Cloud Functions | 60 Min. for HTTP functions, 10 Min. for event-driven functions | 16GB Max function memory | Node JS, Python, Java, Ruby, Go, PHP, .Net | Yes |
| Azure | Azure Function | 5 Min., 10 Min. or unlimited based on plan | Runs on Linux or Windows | C#, Javascript, Java, Python, PowerShell, TypeScript | Yes |
| Digital Ocean | Digital Ocean Functions | 5 Sec | NA | Go, Node JS, PHP, Python | Yes |
| Linode | NA | NA | NA | NA | NA |
| Alibaba Cloud | Function Compute | 600 Sec or pay as you go service | 3072 MB max RAM with 100 MB package limit | Node JS, Python, Java, PHP, C#, Go, Ruby, Lua | Yes |
| Oracle | Oracle Functions | 300 Sec Max, 2048 GB Max | Powered by Fn Project open-source engine | Java, Python, Node, Go, Ruby, C# | Yes |
| IBM Cloud | IBM Cloud Functions | 600000 ms, 2048 MB | Code size limit 48 MB | Node JS, Swift, Java, Python, Ruby, Go, PHP, .NET | Yes |

**Table 5.** Popular service provider for geospatial database. Remote sensing and GIS projects mostly work with geo-spatial database and most providers support to the geospatial database Service providers have not been ranked or kept in any order based on any particular feature.

| Service Provider | Geo-spatial DB Name |
|---|---|
| AWS | Amazon Athena, Document DB |
| GCP | Big Query |
| Azure | Azure Cosmos DB |
| Digital Ocean | None |
| Linode | None |
| Alibaba Cloud | Apsara DB |
| Oracle | Spatial Database |
| IBM Cloud | Informix Server |

Computing instance means remotely available computing device i.e. server (Table 2). Providers provide low computing as well as high computing servers. User can use this computing instance to develop, test, and run applications, models, or products. The performance of the executed application depends on the system configuration. Depending

upon the needs and requirements, we can select low computing instances or high-speed computing instances. In addition, service providers have self-managed service. In this service, the user need not worry about the system configuration. The provider's platform automatically increases or decreases the user's system capacity as required [9]. This increases user's development process and reduces the system maintenance time. The platform also provides technical support where the requests or queries of the user is fulfilled. When a user develop an application, the user uses service provider's services to execute application and store information. The application needs file storage services which is cost effective to store or retrieve information [10]. File storage is about the data including data upload, download, transfer and storage. There could be multiple options provided by service provider on how data could be stored. One option is data can be stored based on how frequently the data needs to be accessed. Users can decide about the file storage based on 3 essential parameters like file type, stored file query frequency, necessity of caching files.

### 1.2. Load balancing, Scaling and Deployment

One way of increasing application capacity is to deploy more server and application in it. While a server has limited resource capacity and can process in limitation, a load balancer works to divide the user request to multiple server [24]. Round robin is the most common algorithm to distribute the work process in load balancer. Service provider provides this facility where the application is deployed to multiple server and the rest of the process is handled by the load balancer itself.

In auto scaling, the capacity of the server is increased. Assuming the running server is crossing the provided threshold, the service provider platform auto increases the resource capacity. For example, if the server has 1GB of RAM, the new RAM of scaled server is set greater than 1 GB of RAM. The resource of the scaling system is increased based on the limit user have set for that specific resource.

During the application development, the application needs to be continuously developed and deployed. Using the auto deployment service developer can make sure no any steps is missed and deployed correctly using the correct successful build. This service takes the application code from the source, works on to follow process to make server deployable ready and then deploys the application on mentioned server [25].

**Table 6.** Load balancing, Auto scaling, Deployment service provided by popular service providers.

| Service Providers | Load Balancing | Auto Scaling | Auto Deployment |
| --- | --- | --- | --- |
| AWS | Yes | Yes | Yes |
| GCP | Yes | Yes | Yes |
| Azure | Yes | Yes | Yes |
| Digital Ocean | Yes | Yes | Yes |
| Linode | Yes | Yes | Yes |
| Alibaba Cloud | Yes | Yes | Yes |
| Oracle | Yes | Yes | Yes |
| IBM Cloud | Yes | Yes | Yes |

### 1.3. Research grant and training materials

Cloud computing services could be expensive to the researchers or individuals working on self-funded projects. Therefore, most of the discussed service provider also provide research grants to such individuals and researchers and encourage using their platform. Using the services from multiple service providers [12-18] and its training materials, frequently usable applications has also been developed [19-23],[26-27 ]. Table 6 gives more details about the grant and support information. In addition, there are training materials and course available for the beginners to learn about the cloud computing services. Table 7 provide more information about it.

**Table 7.** Research grants provided by the popular service provider.

| Service Provider | Research Grant Availability (Credits) |
| --- | --- |
| AWS | Upto 5000 Hrs |
| GCP | Upto $5000 |
| Azure | Based on research and application |
| Digital Ocean | Based on research and application |
| Linode | NA |
| Alibaba Cloud | NA |
| Oracle | Starting from $1000 |
| IBM Cloud | Based on research and application |

**Table 8.** Training materials and courses provided by the popular service providers.

| Service Providers | Documentation and Tutorials | Link |
| --- | --- | --- |
| AWS | Yes | https://docs.aws.amazon.com/index.html |
| GCP | Yes | https://cloud.google.com/docs/ |
| Azure | Yes | https://docs.microsoft.com/en-us/azure/ |
| Digital Ocean | Yes | https://docs.digitalocean.com/ |
| Linode | Yes | https://www.linode.com/docs |
| Alibaba Cloud | Yes | https://arthas.aliyun.com/doc/en/ |
| Oracle | Yes | https://docs.oracle.com/en/ |
| IBM Cloud | Yes | https://cloud.ibm.com/docs/ |

## 2. Conclusion

This paper provides computing information, storage facilities and serverless architecture available within cloud computing domain that can be useful while selecting the appropriate cloud computing platform. The information offered in this study will be helpful for researchers looking to advance cloud design as well as developers looking to deploy their content on multiple service providers. This paper provides baseline information to the users willing to learn about the cloud computing scope and it can be a touchstone for the researchers/users interested in learning about the cloud service usage design or architectural samples.

Author Contributions: Conceptualization, Pranish Dahal, Ritika Prasai; methodology, Pranish Dahal; data cleaning Pranish Dahal, Ritika Prasai; writing—original draft preparation, Pranish Dahal, Ritika Prasai. All authors have read and agreed to the published version of the manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest.

# References

[1]  "Cloud Computing: An Overview | SpringerLink." https://link.springer.com/chapter/10.1007/978-3-642-10665-1_63 (accessed Jul. 06, 2022).

[2]  T. Dillon, C. Wu, and E. Chang, "Cloud Computing: Issues and Challenges," in *2010 24th IEEE International Conference on Advanced Information Networking and Applications*, Apr. 2010, pp. 27–33. doi: 10.1109/AINA.2010.187.

[3]  Y. Altintas, C. Brecher, M. Weck, and S. Witt, "Virtual Machine Tool," *CIRP Ann.*, vol. 54, no. 2, pp. 115–138, Jan. 2005, doi: 10.1016/S0007-8506(07)60022-5.

[4]  G. McGrath and P. R. Brenner, "Serverless Computing: Design, Implementation, and Performance," in *2017 IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW)*, Jun. 2017, pp. 405–410. doi: 10.1109/ICDCSW.2017.36.

[5]  S. Chaisiri, B.-S. Lee, and D. Niyato, "Optimization of Resource Provisioning Cost in Cloud Computing," *IEEE Trans. Serv. Comput.*, vol. 5, no. 2, pp. 164–177, Apr. 2012, doi: 10.1109/TSC.2011.7.

[6]  F. Lombardi and R. Di Pietro, "Secure virtualization for cloud computing," *J. Netw. Comput. Appl.*, vol. 34, no. 4, pp. 1113–1122, Jul. 2011, doi: 10.1016/j.jnca.2010.06.008.

[7]  P. Maresova, V. Sobeslav, and O. Krejcar, "Cost–benefit analysis – evaluation model of cloud computing deployment for use in companies," *Appl. Econ.*, vol. 49, no. 6, pp. 521–533, Feb. 2017, doi: 10.1080/00036846.2016.1200188.

[8]  M. T. Amron, R. Ibrahim, and S. Chuprat, "A Review on Cloud Computing Acceptance Factors," *Procedia Comput. Sci.*, vol. 124, pp. 639–646, Jan. 2017, doi: 10.1016/j.procs.2017.12.200.

[9]  A. Gandhi, P. Dube, A. Karve, A. Kochut, and L. Zhang, "Adaptive, Model-driven Autoscaling for Cloud Applications," 2014, pp. 57–64. Accessed: Jul. 06, 2022. [Online]. Available: https://www.usenix.org/conference/icac14/technical-sessions/presentation/gandhi

[10] A. Reichman, "File Storage Costs Less In The Cloud Than In-House," p. 15, 2011.

[11] I. Baldini *et al.*, "Serverless Computing: Current Trends and Open Problems," in *Research Advances in Cloud Computing*, S. Chaudhary, G. Somani, and R. Buyya, Eds. Singapore: Springer, 2017, pp. 1–20. doi: 10.1007/978-981-10-5026-8_1.

[12] Dr. R. P. S. Mr. A. C. Sekhar, "IRJET-A WALK THROUGH OF AWS (AMAZON WEB SERVICES)", Accessed: Jul. 06, 2022. [Online].                                                  Available: https://www.academia.edu/13017573/IRJET_A_WALK_THROUGH_OF_AWS_AMAZON_WEB_SERVICES_

[13] E. Bisong, "An Overview of Google Cloud Platform Services," in *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*, E. Bisong, Ed. Berkeley, CA: Apress, 2019, pp. 7–10. doi: 10.1007/978-1-4842-4470-8_2.

[14] M. Copeland, J. Soh, A. Puca, M. Manning, and D. Gollob, "Microsoft Azure and Cloud Computing," in *Microsoft Azure: Planning, Deploying, and Managing Your Data center in the Cloud*, M. Copeland, J. Soh, A. Puca, M. Manning, and D. Gollob, Eds. Berkeley, CA: Apress, 2015, pp. 3–26. doi: 10.1007/978-1-4842-1043-7_1.

[15] "IBM Cloud Computing Powering a Smarter Planet | SpringerLink." https://link.springer.com/chapter/10.1007/978-3-642-10665-1_62 (accessed Jul. 06, 2022).

[16] P. Paul, P. S. Aithal, R. Saavedra M, P. S. B. Aremu, and P. Baby, "Cloud Service Providers: An Analysis of Some Emerging Organizations and Industries." Rochester, NY, May 25, 2020. Accessed: Jul. 06, 2022. [Online]. Available: https://papers.ssrn.com/abstract=3613033

[17] *Practical Oracle Cloud Infrastructure*. Accessed: Jul. 06, 2022. [Online]. Available: https://link.springer.com/book/10.1007/978-1-4842-5506-3

[18] G. Zhang and M. N. Ravishankar, "Exploring vendor capabilities in the cloud environment: A case study of Alibaba Cloud Computing," *Inf. Manage.*, vol. 56, no. 3, pp. 343–355, Apr. 2019, doi: 10.1016/j.im.2018.07.008.

[19] M. Amani *et al.*, "Google Earth Engine Cloud Computing Platform for Remote Sensing Big Data Applications: A Comprehensive Review," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 13, pp. 5326–5350, 2020, doi: 10.1109/JSTARS.2020.3021052.

[20] H. Tamiminia, B. Salehi, M. Mahdianpari, L. Quackenbush, S. Adeli, and B. Brisco, "Google Earth Engine for geo-big data applications: A meta-analysis and systematic review," *ISPRS J. Photogramm. Remote Sens.*, vol. 164, pp. 152–170, Jun. 2020, doi: 10.1016/j.isprsjprs.2020.04.001.

[21] Prasai, R. (2022a). An open-source web-based tool to perform spatial multicriteria analysis. *International Journal of Multidisciplinary Research and Growth Evaluation*, 297–301. https://doi.org/10.54660/anfo.2022.3.3.19

[22] Prasai, R. (2022 b). Spatial Multicriteria Analysis Using a Google Earth Engine Based Web Tool. *Academia Letters*. https://doi.org/10.20935/al5897

[23] R. Prasai, "Using Google Earth Engine for the complete pipeline of temporal analysis of NDVI in Chitwan National Park of Nepal," In Review, preprint, May 2022. doi: 10.21203/rs.3.rs-1633994/v3.

[24] M. Rahman, S. Iqbal, and J. Gao, "Load Balancer as a Service in Cloud Computing," in *2014 IEEE 8th International Symposium on Service Oriented System Engineering*, Apr. 2014, pp. 204–211. doi: 10.1109/SOSE.2014.31.

[25] C. Singh, N. S. Gaba, M. Kaur, and B. Kaur, "Comparison of Different CI/CD Tools Integrated with Cloud Platform," in *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, Jan. 2019, pp. 7–12. doi: 10.1109/CONFLUENCE.2019.8776985.

[26]   Prasai, R. (2022c). Earth engine application to retrieve long-term terrestrial and aquatic time series of satellite reflectance data. *International Journal of Multidisciplinary Research and Growth Evaluation*, 165–171. https://doi.org/10.54660/anfo.2022.3.3.11

[27]   Prasai, R., Schwertner, T. W., Mainali, K., Mathewson, H., Kafley, H., Thapa, S., Adhikari, D., Medley, P., & Drake, J. (2021). Application of Google earth engine python API and NAIP imagery for land use and land cover classification: A case study in Florida, USA. *Ecological Informatics*, *66*, 101474. https://doi.org/10.1016/j.ecoinf.2021.101474