**Introduction**

Google Cloud Platform's portfolio of cloud services isn't as extensive as Amazon's or Microsoft Azure's, but it offers some specialized tools for developers that are hard to find elsewhere. It organizes its cloud services into the following nine categories:

* Compute
* Storage and Database
* Networking
* Big Data
* Internet of Things
* Machine Learning
* Identity and Security
* Management Tools
* Developer Tools

Google boasts that its cloud services have customer-friendly pricing, and it claims that its prices are "on average 60% less for many compute workloads compared to other cloud providers.

**Problem Definition**

Amazon launched its cloud computing service in 2006. 2 years later, Google got in the game with its own cloud service. In April 2008, [Google announced a preview release of App Engine](http://googleappengine.blogspot.com/2008/04/introducing-google-app-engine-our-new.html), a developer tool that allowed users to run their web applications on Google infrastructure. The goal of App Engine was to:

“make it easy to get started with a new web app, and then make it easy to scale when that app reaches the point where it’s receiving significant traffic and has millions of users.”

In an effort to gain feedback and make improvements on this preview release, App Engine was made available to 10,000 developers on a first-come, first-served basis. Developers could run apps within the following parameters: 500 MB of storage, 200 million megacycles of CPU per day, and 10 GB of bandwidth per day.

Finally, in November 2011, Google pulled App Engine out of preview mode and dubbed it an official, fully supported Google product. 10 years after the release of this first product, Google has added many more services under the Google Cloud Platform umbrella. It has become one of the top public cloud vendors in the world.

**Objectives**

The purpose of this paper is to explain the working of Cloud service offered by Google Cloud Platform and its impact on the users. By choosing GCP among many Cloud services I seek to understand how the technology works and the reasons behind its success among its users. We will look into how GCP acquire its content and the business model of the service.

**Limitations**

## **.bashrc content**

* + Cloud Shell may produce errors if the .bashrc file is modified or deleted.
* **Slow connection performance**
  + Connecting to a Cloud Shell for the very first time involves creating your home disk and can currently take up to 25 seconds. Subsequent connections to existing virtual machine instances take about five seconds. However, if you haven't used Cloud Shell for a week, connection performance will be slower since the home directory needs to be restored from an archive.
* **Browser support**
  + Cloud Shell supports the latest versions of Google Chrome, Mozilla Firefox, Microsoft Edge, Microsoft Internet Explorer 11+ and Apple Safari 8+. Safari in private browser mode is not supported.
* **Large file transfer**
  + File transfer will sometimes be slow for large files. It's recommended to use gcloud alpha cloud-shell scp or the editor to transfer large files.

**Methodology**

With the recent release of Cloud run, it's now even easier to deploy serverless applications on [Google](https://cloud.google.com/) Cloud Platform (GCP) that are automatically provisioned, scaled up, and scaled down. But in a serverless world, being able to ensure your service meets [t](https://12factor.net/)he twelve factors is paramount.

The Twelve-Factor App denotes a paradigm that, when followed, should make it frictionless for you to scale, port, and maintain web-based software as a service. The more factors your environment has, the better.

## **The Twelve Factors**

* **CODEBASE**

Each service you intend to deploy on Cloud Run should live in its own repository, whatever your choice of source control software. When you want to deploy your service, you need to build the container image, then deploy it.

* **DEPENDENCIES**

Explicitly declare and isolate dependencies.

Since Cloud Run is a Bring-Your-Own container environment, you can declare whatever you want in this container, and the container encapsulates the entire environment. Nothing escapes, so two containers won't conflict with each other.

When you need to declare dependencies, these can be captured using [environment variables](https://cloud.google.com/run/docs/configuring/environment-variables), keeping your service stateless

## **CONFIG**

*Store config in the environment*

Yes, Cloud Run stores configuration in the environment by default.Your code goes in your container, and configuration is captured in the specification of your Service. Configuration includes for example the amount of memory, CPU or [environment variables](https://cloud.google.com/run/docs/configuring/environment-variables). These can be declared when you create the service, in the Optional Settings.

* ***Backing Services***

*Treat backing services as attached resources*

Much like you would connect to any external [database](https://cloud.google.com/run/docs/configuring/connect-cloudsql) in a containerized environment, you can connect to a plethora of different hosts in the [GCP universe](https://cloud.google.com/run/docs/using-gcp-services" \l "services_and_tools_recommended_for_use).

* **BUILD, RELEASE, RUN**

***Strictly separate build and run stages***

## **PROCESSES**

*Execute the app as one or more stateless processes*

## ***PORT BINDING***

*Export services via port binding*

*Cloud Run follows the modern architecture best practices and each Service must expose themselves on a port number, specified by the [PORT environment variable](https://cloud.google.com/run/docs/reference/container-contract" \l "port).*

*This is the fundamental design of Cloud Run: any container you want, as long as it listens on port 8080.*

## ***CONCURRENCY***

*Scale out via the process model*

*Concurrency is a* [*first-class factor in Cloud Run*](https://cloud.google.com/run/docs/about-concurrency)*. You declare what the maximum number of* [*concurrent requests*](https://cloud.google.com/run/docs/configuring/concurrency) *your container can receive. Using this maximum and other factors, Cloud Run* [*will automatically scale*](https://cloud.google.com/run/docs/about-concurrency) *by adding more container instances to handle all incoming requests.*

## ***DISPOSABILITY***

*Maximize robustness with fast startup and graceful shutdown*

*Since Cloud Run handles scaling for you, it's in your best interest to ensure your services are the most efficient they can be. The faster they are to startup, the more seamless scaling can be.*

## ***DEV/PROD PARITY***

*Keep development, staging, and production as similar as possible*

* ***LOGS***

*Treat logs as event streams*

*Cloud Run uses Stackdriver Logging* [*out of the box*](https://cloud.google.com/run/docs/error-reporting)*.* [*The "Logs" tab on your Cloud Run service view*](https://cloud.google.com/run/docs/logging) *will show you what's going on under the covers, including log aggregation across all dynamically created instances*

## ***ADMIN PROCESSES***

*Run admin/management tasks as one-off processes*

*Administration tasks are outside the scope of Cloud Run. If you need to do any project configuration, database administration, or other management changes, you can perform these tasks using the* [*GCP Console*](https://console.cloud.google.com/)*,* [*gcloud CLI*](https://cloud.google.com/sdk/gcloud/)*, or* [*Cloud Shell*](https://cloud.google.com/shell/docs/)*.*

**Business Model**

* Google Cloud platform offers wide range or user
* It provides following infrastucture
  + Data management
  + Application development
  + Smart business analytics and AI
  + Productivity and work Transformation

**The future of Cloud Computing**

Businesses nowadays are seeking innovative ways to grow and accomplish their business goals. With the help of cloud computing, this business will keep on growing in the future. Cloud computing is powerful and expansive and will continue to grow in the future and provide many benefits. Cloud computing is extremely cost-effective and companies can use it for their growth. The future of cloud computing is bright and will provide benefits to both the host and the customer. One should keep in mind that the owner of the company should be familiar with the latest development taking place in Cloud technology.

**Conclusion**

Today, we can connect everything digitally to Cloud Computing. It provides a whole new world of jobs, applications, services, and platforms. We can see the future of Cloud computing as a combination of cloud-based software products and on-premises compute which will help to create hybrid IT solutions.The modified cloud is scalable and flexible, which will provide security and control over data center. One of the integral parts of cloud computing will be the organized process and a better way of processing data. [Cloud has many features](https://data-flair.training/blogs/features-of-cloud-computing/), which makes it’s future brighter in the IT sector

**Citations**

* [**https://cloud.google.com/**](https://cloud.google.com/)
* [**https://linuxacademy.com/**](https://linuxacademy.com/)