

Comparison of Several PaaS Cloud Computing Platforms

Amar Deep Gorai¹, Dr. Birendra Goswami²

¹Guru Nanak College, Dhanbad, ²Usha Martin Academy, Ranchi

Abstract— Today, the question is less about whether or not to use Platform as a Services (PaaS), but rather which providers to use. PaaS is a computing platform that abstracts the infrastructure, OS, and middleware to drive developer productivity. PaaS offerings are “polyglot” and “polyhost”. Selection of Platform as a Service provider is an important process because an ideal vendor will be able to continue to partner with company as company grows. There are many components to be consider while selecting PaaS vendor like Scalability, Availability, Manageability, Performance, Security, Accessibility, Billing At a high-level a PaaS helps organizations, specifically by providing a fast and scalable way to host applications in the cloud.

Key words: Amazon AWS, Benefits of Platform as a Service, Comparison of Platform as a Service, Engine Yard, Google App Engine, Heroku, RedHat OpenShift, Windows Azure Cloud Services

I. INTRODUCTION

Cloud Computing is one of the hot topics of the moment and everyone has an opinion on it. The term ‘Cloud’ covers a number of deployment scenarios, including PaaS (Platform as a Service), SaaS (Software as a Service) and IaaS (Infrastructure as a Service). The computing resources that we need grown up due to the use of computers in our day-to-day life has increased. It is costly to buy a mainframe computers, it became important to find the alternative ways to get the greatest return on the investment, allowing multiple users to share among both the physical access to the computer from multiple terminals and to share the CPU time, eliminating periods of idleness, which is time-sharing. The origin of the term ‘cloud computing’ is unclear, but it appears to derive from the way of drawings of stylized clouds to denote networks in diagrams of computing and communications systems. Cloud computing is a paradigm shift in which computing is moved away from personal computers and even the individual enterprise applications to a ‘cloud’ of computers. “Cloud computing is where data and services reside in massively scalable data centres in the cloud and can be accessed from any connected devices over the internet”. In cloud services users are given access to application software and databases. The cloud service offering companies manage the infrastructure and platforms on which the applications run. Software as a services is sometimes referred to as on-demand software and is usually priced on a pay-per-use basis. Providers generally charge price using a subscription fee.

The Platform as a Service is a way to rent hardware, operating systems, storage and network capacity over the Internet. The service delivery model allows and customer to rent virtualized servers applications or developing and testing new ones. PaaS is a proven model for running applications without the hassle of maintaining the hardware and software infrastructure at your company. PaaS has several advantages for developers. Operating system features can be changed and upgraded frequently with PaaS. Geographically distributed development teams can work

together on software development projects. Services can be obtained from diverse sources that cross international boundaries. Initial and onging costs can be reduced by the use of infrastructure services from a single vendor rather than maintaining multiple hardware facilities that often perform duplicate functions or suffer from incompatibility problems. PaaS involves some risk of “look-in” if offerings require proprietary service interfaces or development languages. Flexibility of offerings may not meet the needs of some users whose requirements rapidly evolve. Platform as a Service offering, usually depicted in all-cloud diagrams between the SaaS layer above it and the IaaS layer below, is a broad collection of application infrastructure services including application platforms, integration, business process management and database services. PaaS offers “auto-scale” for resources that are available for application. It’s one of the huge advantages of PaaS.

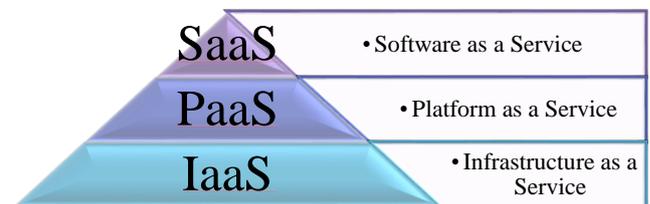


Fig 1 : Stack of Cloud Computing

There are many benefits of Platform as a Services:

- **Physical infrastructure not required:** virtual infrastructure is available on rent which saves cost. Users need not to purchases hardware and employ the expertise to manage it. So Users can focus on the development of application.
- **Cost Saving:** Being a cloud offering, it follows cloud service model – pay per use/pay per go pricing model, where there is no need to buy the software, middleware or full year license. There is no need to buy all the system, software, platforms, tools and kits needed to build, run and deploy the application. Users can rent them for the period for which services will be used. It changes the cost structure form Capital expense (Capex) to Operational Expense (Opex) for an enterprise.
- **Development possible for non-experts:** with PaaS offering anyone can develop an application. This is possible through web browser utilizing one-click functionality. It provides Integrated Development Environment (IDE) to deploy, test. Host and maintain application in the same environment. Like WordPress is one-click blog software.
- **Flexibility:** customers can create a platform that suits their specific requirements and control tools that are installed within that platforms. They can ‘pick and choose’ necessary features.
- **Adaptability:** Features can be changed if circumstances dictate that they should. Built in scalability and elasticity to provide same efficiency and experience irrespective of load and usage.

- **Access form anywhere:** only internet connection and web browser are required, developers can work on same application build together from several location. It provides services required to support the complete life cycle of building and delivering web applications and services on the internet.
- **Match for agile software development methodologies:** an agile software development methodology is based on iterative and incremental development which may require iterations in need of software and other middleware platforms with progress and hence PaaS is the right match for agile application development methodology.
- **Security:** data security and backup and recovery is provided.

II. PAAS CLOUD COMPUTING PLATFORMS PROVIDERS

A range of PaaS vendors exist on the market, from those allowing customization of existing software programs to fully functioning development and deployment environment. Some vendors provides add-on development environments in conjunction with their software programs. Where some other include standalone development packages that provide customized programming environments. Application delivery platforms focus on supporting deployment of programs to customers, and may not include development features such as debugging and testing. More comprehensive packages offer end-to-end support from the programming environment to serving up software to the customers. It's important to evaluate the full range of tools and services available from vendors, and look at their potential to scale support throughout the life of project. Selection of Platform as a Service provider is an important process.

A. Google App Engine

Google App Engine is designed for distributed web applications and developers using Java, Python, PHP and Go. The Java environment supports other languages that make use of the JRE and there is a SDK for each of the four main supported languages as well as a plugin for Eclipse. The PaaS offers managed infrastructure and runtime environments that are guaranteed to scale, but only if the applications fit the restrictions of Google App Engine. The Data-store, a transactional, schema-less data store based on key-value pairs, handles the complex management of data that's accessible to multiple machine instances.



B. Windows Azure Cloud Services

With Windows Azure, Microsoft is blurring the lines between IaaS and PaaS. Windows Azure Cloud Services supports .NET, Node.js, PHP, Python, Java and Ruby. In addition to software development kits, developers can use Visual Studio for creating and deploying applications. Developers can choose between a SQL Database, Tables and Blobs when it comes to persistent storage. Applications

are administered through the Windows Azure dashboard or through a command line interface.



C. Amazon AWS

Although Amazon Web Services is primarily an IaaS, many of the services available in AWS are comparable to PaaS offerings. You can use the platform services available in AWS without having to create or maintain your own application servers. AWS readily supports Java, Python, Ruby, Perl and other languages. Oracle, MySQL and SQL Server can be set up and managed, but AWS offers RDS web service as well, which eliminates database administration tasks. Developers can take advantage of Amazon Elastic Beanstalk for automatic load balancing, auto-scaling, and application health monitoring.



D. Engine Yard

Engine Yard is designed for web application developers using Ruby on Rails, PHP and Node.js who want to take advantage of cloud computing without the operations management responsibility. Engine Yard provides a set of services on top of Amazon AWS. In fact, Engine Yard runs its platform in the Amazon cloud, so the value of the PaaS rests more with orchestration and management than with providing software components. Engine Yard takes care of key operations tasks such as performing backups, managing snapshots, managing clusters, administering databases and load balancing.



E. Red Hat OpenShift

Red Hat OpenShift is based on open source applications and offers a wide variety of languages, databases and components. The PaaS is highly customizable and offered in three forms: OpenShift Online (a cloud-based hosting service), OpenShift Enterprise (a private PaaS that runs in your data center) and OpenShift Origin (the open source application hosting platform). OpenShift automates system administration tasks such as virtual server provisioning, configuration and scaling and supports get repositories for code management.



F. Heroku

Heroku supports Ruby, Python, Java, Scala, Clouture and Node.js. The platform provides abstract computing environments called dynos, which are virtualized Unix-style containers that run processes in isolated environments. Dynos are broadly typed as either web dynos (respond to HTTP requests) or worker dynos (respond to task requests in a queue). Heroku works best with applications that fit well into the Twelve Factor App methodology. Third party applications, called addons, are also available as services within the Heroku platform.



G. AppFog

AppFog is a multi-language, multi-framework PaaS that's a good option for multi-cloud deployments, including private clouds. AppFog supports Java, Ruby, PHP, Python, Node, Scala and Erlang and offers MySQL, PostgreSQL, Redis and RabbitMQ along with third party add-on services. It can be used as a DaaS with ClearDB, MongoHQ, MongoLab, Redis Cloud and Xeround. AppFog is based on the open source Cloud Foundry platform and supports Git, SVN and Mercurial for code management.



III. CONSIDERATIONS AND STEPS TO SELECT PAAS

PaaS is an application platform comprised of an operating system, middleware and other software that allows applications to run on the cloud with much of the management, security, scaling and other stack related headaches abstracted away. This allows organization to focus on two things, customers and developing application. Platform as a Service vendor selection process is important because an ideal vendor will be able to continue to partner with company as company grows. Selection process starts from mapping out own company's needs and determining budget. In next step while searching options online reviews are important, server uptime and ensure statues is valuable, it helps to understand vendors technical support model meets company's needs. If company develops software or has programmers to customize existing software programs, then cloud computing could help to achieve goals on budget and in record time.

Leading PaaS offerings are "polyglot". Means PaaS offers multiple programming languages and frameworks. So developer can code in whatever language he like which can be Java, Ruby, Python, PHP, Perl, and JavaScript etc. developer can develop with new frameworks like play, tried and true systems like java EE. PaaS offering are "polyhost". Means PaaS supports a choice of underlying hosting environments. Whether public cloud, private cloud, local

hypervisor, or bare metal, supporting multiple hosting environments allows the application developer/owner to migrate their application as needed to support increased demand, mitigate cloud outages, address data privacy concerns, or protect themselves from cloud price hikes.

In selection on Platform as a Service vendor some properties are need to be consider these properties are:

- Scalability: PaaS needs to support dynamic provisioning that can increase or decrease resources on demand. PaaS provides the abstraction of cloud infrastructure and the automatic management.
- Availability: PaaS needs to support the fault tolerance ability such that system would not crash on failure. PaaS also needs to provide system resilience by duplicating applications or services.
- Manageability: PaaS needs to support self-management for running applications and services on the cloud platform. PaaS needs to provide automatic control, analysis and measurement for the resource usage.
- Performance: PaaS needs to support the capabilities of distribute jobs to idle resources and remove jobs from the overloading resources. PaaS also needs to distribute data to storage system with load balancing.
- Security: Security is an important characteristic in PaaS. PaaS needs to provide authentication and authorization to differentiate the access rights of different users.
- Accessibility: PaaS needs to provide an interactive interface for consumers to access cloud services or monitor the system status. Consumers could develop and test their applications via web browsers or other thin-clients.
- Billing: bill payment must be according to how many units of resources and services they use which can be:
 - o Input / Output network bandwidth
 - o Management report or warning
 - o CPU time
 - o Storage space
 - o Data Migration

There are some steps and considerations while selecting Platform as a Services:

- Define the business strategy for cloud application: PaaS selection should follow application and business strategy.
- PaaS offerings should provide some essential services:
 - o Application development
 - o Database
 - o Integration
 - o Support
 - o Security
- Determine in advance the importance of application portability: There are two types of PaaS
 - o Portable PaaS offerings are independent of the infrastructure provider. The best offerings in this category are open source and include Cloud Foundry, OpenStack, OpenShift and CloudStack.
 - o Vertically integrated offerings seamlessly combine and IaaS and PaaS and are not portable and include Force.com, Azure and Amazon Web Services.
- Which development frameworks and languages are supported? Is imported to check does PaaS support same .NET development environment which company is using.

It is important that selected PaaS can accommodate preferred development languages and frameworks.

- Availability of experienced resources are important to help designing application in the PaaS. Designing a cloud application is a new skill set.

IV. COMPARISON OF PAAS CLOUD COMPUTING PLATFORMS

Google App Engine, Windows Azure Cloud Services, Amazon AWS, Engine Yard, Heroku, AppFog are available as public Platform as a Service, and RedHat OpenShift is available as private Platform as a Service.

	Languages	Middleware	Framework	Native Services	Add-ons	Infrastructure	Service Year
Google App Engine	4	0	2	3	0	2	4
Windows Azure	6	0	0	0	0	3	5
Amazon AWS	6	5	0	0	0	8	4
Engine Yard	4	7	5	5	21	8	9
Red Hat OpenShift	6	2	4	4	0	0	2
Heroku	8	0	5	1	143	2	6
AppFog	5	2	3	5	21	4	3

Table 1: Comparison Data of Several PaaS Vendors

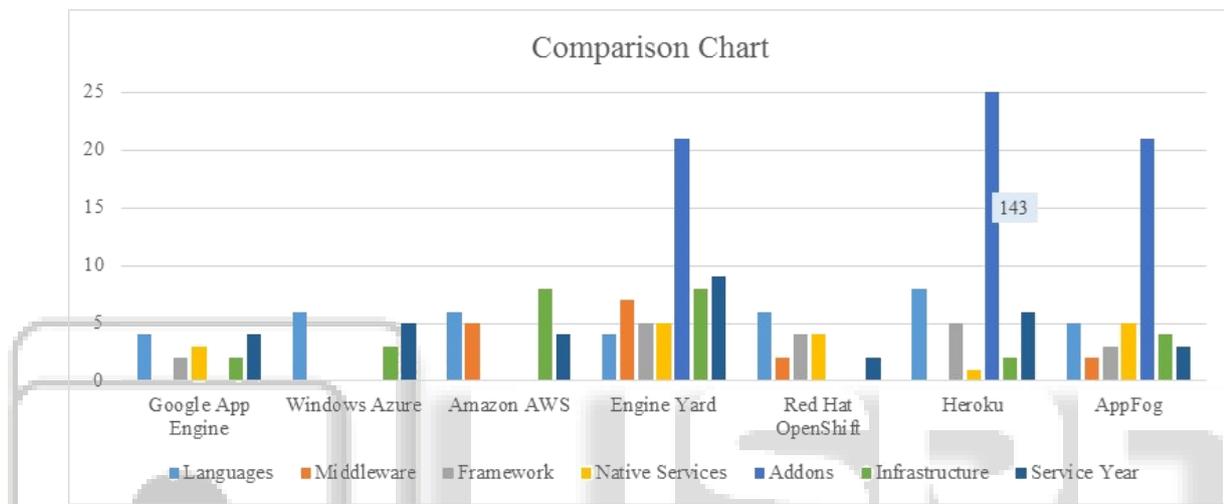


Fig 2 : COMPARISON CHAT OF SEVERAL PAAS VENDORS

RunTimes-Versions	Google App Engine	Windows Azure	Amazon AWS	Engine Yard	Red Hat OpenShift	Heroku	AppFog
Clojure						✓	
Dotnet		✓	✓				
Go	✓						
Groovy						✓	
Java	✓	✓	✓	✓	✓	✓	✓
Node		✓	✓	✓	✓	✓	✓
Perl					✓		
Php	✓	✓	✓	✓	✓	✓	✓
Python	✓	✓	✓		✓	✓	✓
Ruby		✓	✓	✓	✓	✓	✓
Scala						✓	

Fig. 3 : Languages Supported by different PaaS vendors

List of native app supported by different PaaS vendors

- Google App Engine
 - o Google Cloud Datastore
 - o Google Cloud Sql
 - o Google Cloud Storage
- Windows Azure Cloud Services
 - o No Native
- Amazon AWS
 - o No Native
- Engine Yard
 - o Memcached
 - o MySql
 - o Postgresql
 - o Redis
 - o Riak
- Red Hat OpenShift
 - o Jenkins
 - o Mongoddb
 - o Mysql
 - o Postgresql
- Heroku
 - o Postgresql
- AppFog
 - o Mongoddb
 - o Mysql
 - o Postgresql
 - o Rabbitmq
 - o Redis

List of Frameworks supported by different PaaS vendors

Frameworks	RunTime
Google App Engine	
Django	Python
Webapp2	Python
Windows Azure Cloud Services	
Grape	Ruby
Merb	Ruby
Rack	Ruby
Rails	Ruby
Sinatra	Ruby
Amazon AWS	
N/A	N/A
Engine Yard	
N/A	N/A
Red Hat OpenShift	
Django	Python
Drupal	Php
Flask	Python
Rails	Ruby
Heroku	
Django	Python
Flask	Python
Grails	Groovy
Play	Java
Rails	Ruby
AppFog	
Django	Python
Flask	Python
Rails	Ruby

Key Features

Google App Engine
Google App Engine uses a sandbox model which isolates processes thereby reducing the risk that a rogue process on a physical server will disrupt operations of other processes on that server.
Windows Azure Cloud Services
Since Windows Azure is an IaaS and PaaS in one, developers can mix and match IaaS components with PaaS offerings giving you more control.
Amazon AWS
Since AWS is largely an IaaS, there is virtually no limit to the languages, databases or server side technologies you can install and run.
Engine Yard
Dedicated instances with no multi-tenancy at the virtual machine level. More control over your virtual machine instances than other PaaS providers. Integration with both public and private Git repositories.

Red Hat OpenShift
A large number of component options spanning the application stack from front end to backend services. Developers can interface with OpenShift through a web console, the command line or through an integrated development environment.
Heroku
One of the early PaaS providers, Heroku is ideal for quick deployments and fits a wide range of distributed applications.
AppFog
The ability to run your applications on multiple clouds is AppFog's biggest advantage. Those who prefer the VMware platform will also find the PaaS appealing.

Table I: Key Features

Limitations
Google App Engine
Programming languages are limited to Java, Python, Go and PHP.
Windows Azure Cloud Services
Minimalist administration portal.
Amazon AWS
AWS services can require more management overhead than other PaaS options.
Engine Yard
Supported languages are limited to Ruby, JRuby, REE, Rubiniu, Node.js, and PHP.
Red Hat OpenShift
Works well with Git, but non-Git deployments might require additional steps.
Heroku
Cost of addons vary and estimating costs can be challenging as the number of addons increase and loads vary across application components.
AppFog
Currently AppFog does not have persistent file system; other object storage systems can be integrated at an additional cost.

Table II: Limitations

Pricing
Google App Engine
Based on usage. \$0.08/hour per on-demand instance or \$0.05/hour for a reserved instance. The Data-store is billed at \$0.18/GB per month; bandwidth costs \$0.12/GB. Other service costs may apply.
Windows Azure Cloud Services

Based on size of instances running; prices range from \$0.02/hour (768MB of RAM & 1 shared virtual core) to \$0.64/hour (14GB of RAM & 8 virtual cores). Pricing for high memory machines is also available.
Amazon AWS
Complex mixture based on instances, storage, application services and data egress charges. Amazon, however, offers a monthly calculator to help estimate your costs.
Engine Yard
Based on a "pay as you go" model with standard and premium support options. Prices range from \$0.05/hour per instance to \$2.19/hour per instance, depending on your configuration.
Red Hat OpenShift
OpenShift Online pricing is based on the number and types of components (called gears) you deploy. Gear prices range from \$0.02/hour to \$0.10/hour, depending on the size: 512MB (small), 1GB (medium), or 2GB (large). The Silver support plan is \$20/month plus usage costs.
Heroku
Based on the number and size of dynos deployed (\$0.05/hour - \$0.10/hour), the size of the Postgres database and add-ons used.
AppFog
Based on the level of service, add-ons deployed and includes charges for exceeding quotas. Service levels are based on memory instead of CPU. Pricing plans range from \$20/month (2GB RAM) to \$720/month (32GB RAM).

Table III: Pricing

Bonus
Google App Engine
The first 28 instance hours, 1GB of storage and 1GB of inbound/outbound traffic per application are free each day.
Windows Azure Cloud Services
Free 30-day trial with a limit of up to \$200 is available for new users.
Amazon AWS
New users can get 750 hours, 30GB storage and 15GB bandwidth for free with AWS's Free Usage Tier.
Engine Yard
Not Available
Red Hat OpenShift
A limited number of resources are available as a trial; 3 small gears and 1GB of storage per gear are free.
Heroku
Starter databases with up to 10,000 rows are free.
AppFog

Free version allows unlimited apps within 2GB of RAM, 100MB persistent storage (MySQL or PostgreSQL) and up to 8 service instances.

Table IV: Bonus

V. CONCLUSION

PaaS represents the future of software development. In PaaS user need to worry about writing the code for his big idea. Using IDE, command line tools, or a browser interface and he can quickly and easily push app to the PaaS where it will sun on top of an elastic infrastructure that auto-scales to meet the explosive demand of application. As there are many vendors, Providers like Google App Engine, Windows Azure Cloud Services, Amazon AWS, Engine Yard, RedHat OpenShift, and Heroku are considered for enhancing from different dimensions. As a Service market, however, to prove one of the most exciting ones in the cloud space, and there have been several important factors, such as changes in pricing strategies, compatibility operating systems and languages/runtime, supporting services and the entrance of some technology heavy weights.

REFERENCES

- [1] Google App Engine, <https://appengine.google.com/>, Accessed on 9 November 2014
- [2] Windows Azure Cloud Services, <http://azure.microsoft.com>, Accessed on 9 November 2014
- [3] Amazon EC2, <http://aws.amazon.com>, Accessed on 9 November 2014
- [4] Engine Yard <https://www.engineyard.com>, Accessed on 9 November 2014
- [5] Red Hat OpenShift <https://www.openshift.com>, Accessed on 9 November 2014
- [6] Heroku <https://www.heroku.com>, Accessed on 9 November 2014
- [7] AppFog <https://www.appfog.com>, Accessed on 9 November 2014
- [8] AWS Cost Calculator <http://calculator.s3.amazonaws.com/calc5.html>
- [9] Wiki, http://en.wikipedia.org/wiki/Platform_as_a_service, Accessed on 9 November 2014
- [10] Cloud Computing: Analysis of Various Platforms, By Nawsher Khan, A. Noraziah, Elrasheed I. Ismail and Mustafa Mat Deris
- [11] How Cloud Computing Is Changing the World businessweek.com
- [12] The Future of Cloud Computing dartmouth.edu MBA Fellows Program Archive
- [13] Cloud Computing Research by J.E. Cairnes School of Business & Economics
- [14] Evidence-Based User Experience Research, Training, and Consulting by Nielsen Norman Group
- [15] For miscellaneous information: <http://www.cse.buffalo.edu/~bina>
- [16] Comparison of Several IaaS Cloud Computing Platforms by Amar Deep Gorai and Dr. Birendra Goswami, International Journal for Scientific Research & Development, Vol. 2, Issue 09, 2014.