

APP DEVELOPMENT ON THE CLOUD MADE EASY WITH PAAS



This article looks into the benefits of using the Platform as a Service paradigm to develop applications on the cloud. It also compares a few top PaaS providers in the market today.

For a typical software engineer or developer, a platform might be a simple utility, which is used to develop some applications. However, in the cloud, this view may not necessarily hold true. The concept of PaaS or Platform as a Service has come a long way from just providing simple middleware to users.

It now provides a full-fledged, automated application deployment service that can move your code through the different stages of developing software such as development, quality control, testing, production, etc.

Here's a simple definition of PaaS: PaaS is a cloud computing service model that provides scalable application infrastructure and platforms such as application servers, Web and database services, messaging and integration services, etc, on a pay per use basis, without the overhead of buying and managing the underlying software and hardware.

Why PaaS makes sense for application development

There are many benefits of developing applications on the cloud using PaaS, and a few of them are summarised below.

Time and cost savings: Setting up platform-level software is a complex and time consuming activity. PaaS abstracts a lot of the complexities of managing low level infrastructure and application dependencies so that the software developers simply have to upload their code and start running it without having to worry about procuring and configuring additional servers, software licenses, etc. No upfront payments or contracts are required by a PaaS provider. You simply use the platform and pay for what you use.

Scalability and availability: A PaaS platform also provides on-demand scalability and high availability for your applications. If there is an increased workload on your application, then the PaaS platform automatically can add more servers and sync load balancers to maintain the optimal efficiency of

your running application. Once the load or requests on your application reduce, the additional servers are de-provisioned automatically.

Easy upgrades: A typical application deployment may rely on a number of components that individually need to be tracked, maintained, upgraded and re-integrated over time. The PaaS provider abstracts the management, timely security and patch upgrades of the PaaS platform so that the users do not waste time and incur risks by doing these tasks themselves.

A vast ecosystem: A PaaS platform provides a rich set of tools, frameworks and UIs for managing and building your applications on the cloud. A developer has the freedom to choose from a variety of development languages such as Java, .NET, Node.js, PHP, Python, Ruby, etc, all pre-installed, pre-configured and ready for use.

No vendor lock-in: A PaaS platform provides the freedom of choice to developers to move to different PaaS platforms without being locked in to any vendor.

PaaS providers

There are a variety of PaaS providers in the market today, each providing their own set of tools and frameworks to develop applications. Here's a look at some of the market leaders in this space.

Windows Azure: Windows Azure is a PaaS platform developed by Microsoft to directly compete with other IaaS and PaaS providers such as Amazon Web Services (AWS) and Google App Engine (GAE). Windows Azure provides a rich set of tools and frameworks, which are specific to both

Microsoft and third parties, that can help developers build their applications on the cloud.

Additionally, it also provides media encoding services, multi-tier applications, and automated deployment services using Git and Eclipse.

Windows Azure also provides SQL Database as a service in another offering called SQL Azure Database. This can help developers integrate their applications with a scalable database that supports Microsoft's Active Directory components, Microsoft System Center, Hadoop Processing Framework, etc.

Azure provides the Blob store and table-like structures to store your application data. It has a REST, XML and http-based API that can be used by developers to extend their application's functionality with the Azure framework.

The pricing is simple and also offers billing on a pay-as-you-go model as well as pre-paid plans.

Amazon Elastic Beanstalk: Amazon Elastic Beanstalk is a PaaS-like service from Amazon Web Services (AWS), a major IaaS provider. Beanstalk provides users the ability to quickly deploy their applications and manage them on the AWS Cloud. It provides an easy to use Web UI with which developers can upload their code as a WAR file, tune in a couple of necessary parameters, and finally launch their application on AWS. Beanstalk will

automatically provision the virtual machine instances, monitor the workloads of your application and balance load across the instances effectively.

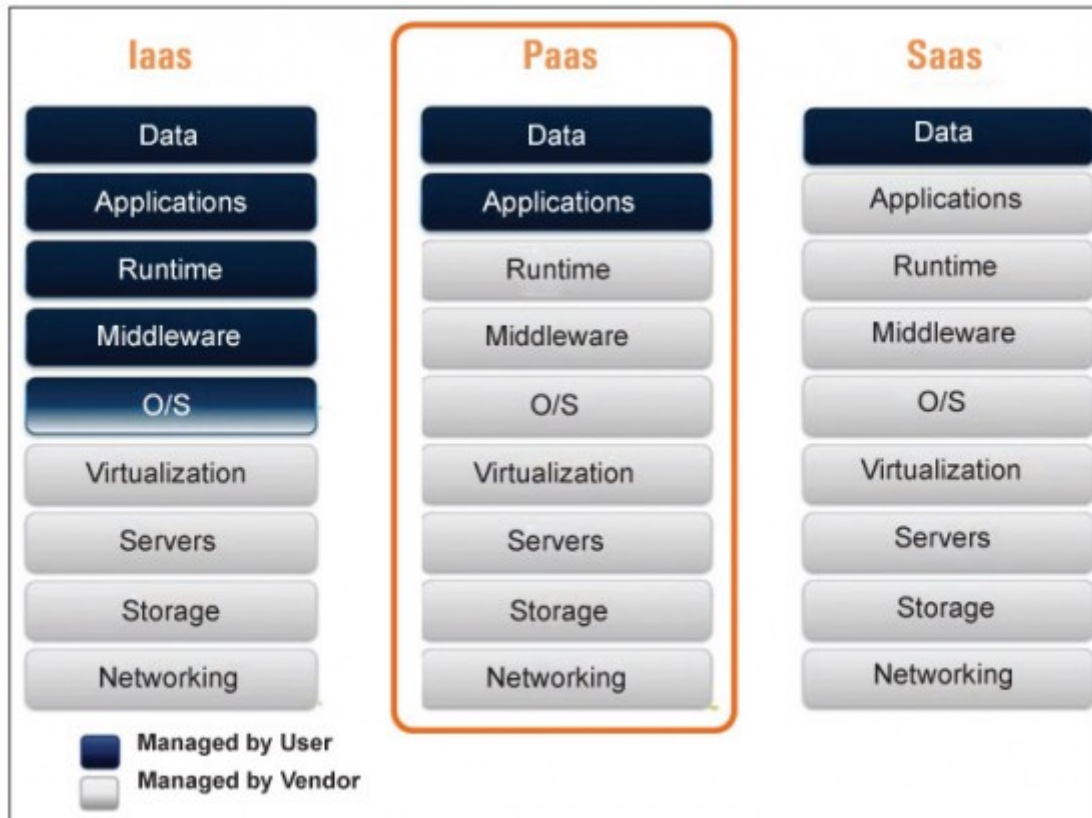


Figure 1: PaaS stack

Beanstalk provides users complete control over their AWS resources that power the application. Users have the ability to browse through the application log files, monitor the application health and even pass the necessary run time values to their applications.

Users also have the freedom to directly publish their code on Elastic Beanstalk using tools such as Visual Studio, Eclipse IDE and Git.

Since AWS offers loosely coupled services, Elastic Beanstalk can be integrated with other AWS services such as Simple Notification Service

(Amazon SNS), Simple Storage Service (Amazon S3), Auto-Scaling, Elastic Load Balancer (ELB), etc.

Elastic Beanstalk currently can deploy applications written in .NET, Java, Node.js, Ruby, Python and PHP.

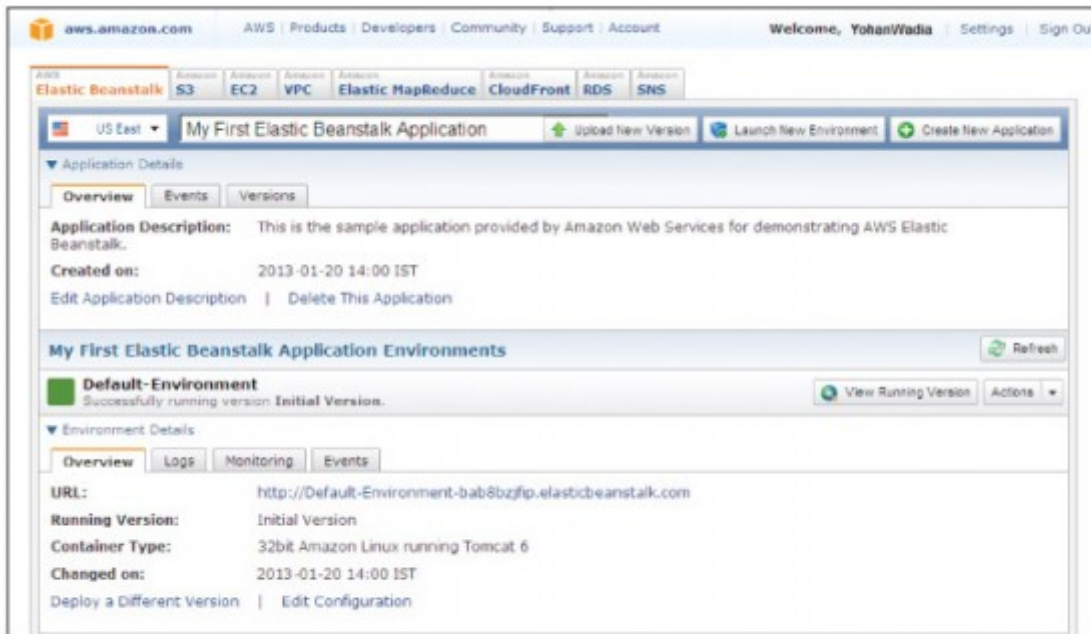


Figure 2: Elastic Beanstalk management console

There is

no additional charge for using Amazon Elastic Beanstalk. Users just have to pay for the underlying AWS resources that they consume, such as EC2, S3, ELB, etc.

Google App Engine: Google App Engine (GAE) is a PaaS offering that enables users to host their Web applications on Google's infrastructure, i.e., Google's globally present data centres in a 'Sandbox' like environment. GAE offers auto-scaling of resources, and load balancing that enables your Web application to scale automatically when the workloads increase.

Currently, only Java and Python are supported as the programming languages; however, GAE does provide an experimental API called 'Go', which enables users to program using additional languages such as Groovy, Scala and PHP.

Similar to Windows Azure, GAE also provides additional means for storing your data using three main services:

- ♣ *App Engine Datastore*: A NoSQL object store that comes bundled with a query engine.
- ♣ *Google Cloud SQL*: A highly scalable relational SQL database service that is very similar to MySQL.
- ♣ *Google Cloud Storage*: A simple storage container that can store files and objects pertaining to your application.

GAE is priced similar to Amazon's Elastic Beanstalk. It is a free utility. You only have to pay for the resources that you consume.

Cloudify: Cloudify is another example of an open source PaaS platform. It relies on a 'recipe-based model to deploy applications on any cloud-based platform without having to undergo changes in the application code. The 'recipe' can be thought of as a template or blueprint for installing, configuring and monitoring the entire application stack. These 'recipes' are designed to be cloud-agnostic and, hence, can be executed on multiple cloud platforms without any major changes made to them.

Cloudify currently supports and provides recipes for a variety of application servers such as Tomcat, JBoss and Oracle Web Logic; for scalable databases such as Cassandra and MongoDB, and even for a host of private and public clouds including Amazon Web Services, Openstack, Cloudstack, etc.

Like most PaaS providers, Cloudify also provides automated auto-scaling and dynamic load balancing of resources.

Open Shift: Open Shift is a PaaS offering from Red Hat and it primarily comes in two variants an enterprise PaaS that is meant to be deployed on private clouds, and the Web-based Open Shift that allows you to launch your applications on a Red Hat provided cloud. There is an open source version available on GitHub as well, called Open Shift Origin.

Getting started with Open Shift is fairly easy. Developers can create an application using Open Shift's command line console or an IDE, code the application and then push it on to Open Shift.

Once published, Open Shift will take care of the application's underlying services and even scale the application automatically in case the workloads increase.

Open Shift supports a variety of development languages such as Java, Ruby, PHP, Python, Perl, Node.js as well as databases such as MySQL, PostgreSQL and MongoDB.

The Open Shift platform also provides a facility for continuous integration of code and its release management, using Jenkins. Once the application code is submitted, Jenkins automatically builds it, performs testing on the code and

publishes it, depending on the test results. This enables you to develop applications in a more automated and streamlined fashion.

There are many more leading PaaS providers in the market today Salesforce.com, Heroku, AppFrog, AppScale, etc each one providing a different set of services and tools for developers to build and host their applications on. Their growing popularity proves that PaaS is a faster, cheaper, more reliable and secure way to develop applications, and is here to stay.

Source : <http://www.opensourceforu.com/2013/12/app-development-cloud-made-easy-paas/>